American Journal of Computational Linguistics *Microfiche 36*

PROCEEDINGS.

13TH ANNUAL MEETING

ASSOCIATION FOR COMPUTATIONAL LINGUISTICS

5: MODELING DISCOURSE AND WORLD KNOWLEDGE II. AND TEXT ANALYSIS

Timothy C. Diller, Editor

Sperry-Univac
St. Paul, Minnesota 55101

PREFACE

The fifth and final ACL session was split into two subsessions: one continued the treatment of discourse structure and general knowledge begun in session 4; the other provided a look at several automated text analysis systems. Georgette Silva kindly chaired both subsessions.

Only five of the six talks given are represented in this Proceedings. The paper detailing Salton's talk on automatic indexing was far too extensive to be included on this fiche and hence will be published separately. The paper by Klappholz and Lockman discusses the problems involved in the resolution of cross-sentential reference and sketches an algorithm for their solution. (Note the closely related paper by Deutsch in Session 4.) Rosenschein addresses the problem of restricting the generation of inferential propositions given a set of beliefs and proposes a structural constraint upon inferencing. Beckles et al. present a man-machine approach to the description of idiolect variations in an environment extraordinarily complex linguistically and sociologically. Brill and Oshika describe a set of programs which permit both batch and interactive processing of orthographic and phonological strings to provide information on frequency, contextual variation, and associational relations. Anderson, Bross, and Sager present a theory of linguistic compression in written texts and describe the results of an implementation of that theory.

Timothy C. Diller, Program Committee Chairman

TABLE OF CONTENTS

SESSION 5A: MODELING DISCOURSE AND WORLD KNOWLEDGE II

SESSION 5B: TEXT ANALYSIS

American Journal of Computational Linguistics *Microfiche 36 : 4*

CONTEXTUAL REFERENCE RESOLUTION

DAVID KLAPPHOLZ AND ABE LOCKMAN

Department of Electrical Engineering
and Computer Science
Columbia University
New York, New York 10027

ABSTRACT

With the exception of pronomial reference, little has been written (in the field of computational linguistics) about the phenomenon of reference in natural language. This paper investigates the power and use of reference in natural language. and the problems involved in its resolution. An algorithm is sketched for accomplishing reference resolution using a notion of cross-sentential focus, a mechanism for hypothesizing all possible contextual references, and a judgment mechanism for discriminating among the hypotheses.

The reference resolution problem

The present work began as an attempt to develop a set of algorithms and/or heuristics to enable a primitive-based, inference-driven model of a natural language user (Schank 1972 Rieger 1974) to properly resolve pronomial references across sentence boundaries. The authors quickly realized, however, that the problem of pronomial reference resolution is only a small aspect of a problem which might be termed nominal reference resolution, itself but a small aspect of the problem of the coherence of a text, (or conversation) i. e. the manner in which it "means" more than the logical conjunction of the meanings of its individual constituent sentences.

Examples of the first problem, i. e. pronomial reference resolution are given in sentence sequences 1-4 below.

1. Yesterday some boys from our village chased a pack of wild dogs; the largest one fell into a ditch.
2. The wild dogs which forage just outside our village suffer from a strange bone-weakining disease. Yesterday some boys from our village chased a pack of wild dogs· the largest one broke a leg and fell into a ditch.
3. Yesterday John chased Bill half a block; he was soon out of breath.
4. My friend Bill has an extremely severe case of asthma. Yesterday John chased Bill half a block; he was soon out of breath.

The problem in utterance (text, conversation etc.) excerpts of the above type is that of determining the referents of the various occurrences of the pronouns "one," and "he"

For the moment we simply note that <u>usually preferred</u> referents of the two occurrences of "one" are "boy" and "dog", (examples 1 and 2 respectively) and those of the two occurrences of "he" are "John" and Bill (examples 3 and 4 respectively.)

The more general problem of nominal reference resolution is exhibited in the following annotated excerpt from a recent newpaper article (N. Y. Times 7/15/75, byline Arnold Lubasch); subscripted bracketing of the excerpt is intended only to enable later reference to specific parts of the text.

$_{1}$[Some of the major provisions of $_{2}$[the state's Fair Campaign Code]$_{2}$]$_{1}$ were declared unconstitutional here yesterday oy $_{3}$[a special Federal court]$_{3}$ that assailed $_{4}$[the restrictions on election campaigning] as "repugnant to the right of freedom of speech."

$_{5}$[The three-judge court,]$_{5}$ which was convened to consider a constitutional challenge by three State Assembly candidates last year threw out $_{6}$[$_{7}$[the'code's]$_{7}$ prohibition against attacking any political candidate's race, sex, religion or ethnic background]$_{6}$

$_{8}$[It]$_{8}$ also overtuned $_{9}$[$_{10}$[$_{11}$[the code's]$_{11}$ ban]$_{10}$ on any misrepresentation of a candidate's party affiliation, position on political issues and personal qualifications, including the use of "character defamation" and scurrilous attacks."]$_{9}$

According to $_{12}$[the court's]$_{12}$ 38-page decision, written by $_{13}$[Judge Henry F. Werber]$_{13}$ with the concurrence of $_{14}$[Judges Leonard P. Moore, and Mark A. Constantino]$_{14}$. $_{15}$[$_{16}$[the provisions]$_{16}$ banning misrepresentation]$_{15}$ "cast a substantial chill on the expression of protected speech that are unconstitutionally overbroad and vague."

If newpaper reporters had a bit more sympathy for those of us concerned with natural language processing, the above excerpt might have read as follows:

The state has a Fair Campaign Code.

Some of the major provisions of the state's Fair Campagin Code are provisions which restrict something.

Some of the things restricted by some of the major provisions of the state's Fair Campaign Code which restrict something are activities having to do with election campaigning.

Some of the activities having to do with election campaigning which are restricted by some of the major ,provisions of the state's Fair Campaign Code which restrict something are attacking a political candidate's race, sex, religions or ethnic background and misrepresenting a candidate's party affiliation, position on political issues ...

Last year three state assembly canddiates filed a constitutional challenge to some of the major provisions of the state's Fair Campaign Code which restrict something.

Yesterday a special Federal court declared unconstitutional those of the major provisions of the state's Fair Campaign Code which restrict something ...

The point is that in order for a machine or a human to validly claim to have "understood" the original excerpt he/she/it must be able at the very least to demonstrate that he/she/it has established the following relationships between various items occurring in the excerpt.

(Integers represent subscripted bracketed regments of the original excerpt.)
(i) The identity of 2, 7, and 11
(ii) The identity of 3, 5, 8, and 12
(iii) The fact that 4, 6, 9, and 15 are elements, subsets or parts of 1
(iv) The fact that 13 and 14 are members of 3
and on and on and on. (I. e. a closer analysis of the original excerpt reveals many more relationships which must be established before "understanding" may be claimed.)

If people actually wrote/spoke in the style of the somewhat facetious paraphrase of the original excerpt, the nominal reference problem would be reduced to one of matching lexcial patterns and recognizing a few syntactic cues; to state the obvious, the necessity for more succinct linguistic communication has forced the development of elliptical devices which shift the burden of nominal reference resolution from syntactic analysis to an analysis of the "semantics" of <u>sentences in context</u>. More specifically, nominal references cannot in general be resolved without the use of general semantic information as well as specific world knowledge.

While the fact that syntactic analysis alone is insufficient for understanding is anything but novel, the question of the magnitude of the nominal reference problem and of its solution's crucial dependence upon local context seems to have been little commented upon. (Clark (1975) discusses the problem from a viewpoint different from that of this paper.)

The reader who remains unconvinced by the examples above that local context (and specific world knowledge relating to local context) must play a crucial role in reference resolution is asked to consider the two sentence sequences 5a, 6, and 5b, 6.

5. a. The founding fathers had a difficult time agreeing on how the basic laws governing our country should be framed.

   b. Those foolish people at the country club have spent an incredible amount of time arguing about club rules.

6. The second article of the constitution, for example, was argued about for months before agreement was reached.

In sentence sequence 5a, 6, "the second article" clearly refers to the second article of the constitution of the United States, while in sentence sequence 5b, 6, the reference is to the second article of the constitution of the country club. In each case the only factor involved in resolving the reference is the semantic content of its local context- in this case the meaning of the sentence preceding the one in which the reference occurs.

Since the lexical item "the constitution" appears in the example just considered, a word concerning such proper-noun-like objects is in order. In any language there are lexical items and phrases such as those appearing in 7 below, which, in the absence of compelling alternative, have standard default referents; for example the standard default referents of the items in 7 are the corresponding items in 8 below.

7. a. The constitution
   b. The founding fathers
   c. Wall Street
   d. The establishment
   e. The president
   f. Madison Avenue

8. a. The constitution of the U.S.
   b. The founding fathers of the U.S.
   c. The U.S. business. community (or that part of it residing in New York City.)
   d. Those people who have the power to influence the course of events in the nation etc. etc.
   e. The president of the U.S.
   f. The advertising industry.

In order for textual occurrence of such proper-noun-like objects to be properly handled, their standard default referents must be listed in the lexicon. This is not to say that occurrences of proper-noun-like objects <u>cannot</u> be references to objects occurring previously in the text; rather it is the case that their default options must also be considered as possible referents.

As final examples of the reference resolution problem let us consider sentence sequences 9 and 10 below.

9. The president was shot while riding in a motorcade down one of the major boulevards of Dallas yesterday; it caused a panic on Wall Street.

10. John was invited to tea at the Quimby's last Saturday; he would have loved to go, but he knew he'd be busy then.

In example 9, while the first sentence of the sequence contains a number of noun objects (president, motorcade, boulevards, Dallas) which are potential referents for the occurrence of "it" in the second sentence, none of the these is in fact, the proper referent; rather, the proper referent of "it" is the event (or fact) that "The president was shot while ... ."

In example 10 we have an instance of an adverbial reference ("then") which must be recognized as referring to "yesterday" rather than to some non adverbial object occurring in the first sentence of that example.

Sketch of a Solution

From the point of view of computer implementation, the problem of nominal reference resolution is one of creating tokens for noun objects mentioned in a text, and discovering and encoding the relations, alluded to in the text, which hold between them and various other tokens in memory.

This problem, though certainly not its magnitude or ramifications, was noticed by Rieger (1974) in his poineering implementation of a primitive-based model of a natural language user. Rieger's system, however, suffers from the incredible inefficiency resulting from its need to search all of memory in order to attempt any reference resolution; in addition it will often miss a quite obvious referent entirely, and, in fact, resolves non-pronomial references only accidentally if at all.

Before presenting a sketch of a proposed solution to the nominal reference resolution problem, it would be well to detail more precisely the overall language processing enviornment within which it is meant to operate and of which it is a most necessary part.

First, we assume that a relatively small set, S, of semantic primitives and a logical-calculus-like language, L, for expressing "meanings" are available. The set S and language L must satisfy the following two conditions.

(i) The predicate, function, and constant symbols of L are members of S.

(ii) There is a one-to-one mapping from meanings of (natural language) sentences to formulas of L.

While a set of primitives and a meaning representation language even demonstably close to satisfying the above conditions have yet to be produced, we will, in examples to follow, make use of meaning represen tations; the only claim we will make for them is that the functions served by their constituent constructs must be served by the elements of any adequate system.

In addition to a meaning representation scheme we will assume an encoding of world knowledge of the sort which a "typical" adult might possess, again with the same obvious caveat.

While the question of translation from natural language sentences to meaning representations will not be touched upon here, we will assume sentence-by-sentence translation of the sort exhibited in various examples to follow.

The solution to the reference resolution problem rests in recognizing the fact that reference is an elliptical device, and that the human understander of natural language cannot recapture that which was elided once he is too far from it in the text; in fact, he cannot resolve a reference to a point in the text more than a few-sentences back without going back and pondering it (if he can do so at all). We should note that this is true even in the case in which the referent doesn't actually appear in the text, but appears only in an inference from some statement made in the text. In this latter case - a case which we will discuss only at the very end of this paper the reference is not resolvable (and would not therefore have been made by the creator of the text in the first place).unless the statement from which the inference is made appears shortly before in the text. Though we cannot say precisely how far back is meant by "shortly before," it is certainly no more than a few sentences. For a given sentence, S, appearing in a text we will refer to the sequence of sentences preceding S by no more than the intended distance as the focus of S.

In terms of computer implementation, we will, in the processing of a text (which we conceive of as proceeding sentence-by-sentence), maintain the following focus sets.

(i) The noun-object focus - the set of tokens of all noun objects in the meaning representations of the focus of S (where S is the sentence currently being processed)

(ii) The event focus - a set containing, for every sentence W in the focus of S, the object EVENT(F), where F is the <u>meaning representation</u> of W, and EVENT is a function which maps the meaning of a formula, F, into a noun-like object whose meaning is "the event (or fact) that F"

(iii) The time focus - a set containing takens for all time references (e.g. yesterday, five o'clock, etc.) occurring in the <u>meaning representation</u> of the focus of S.

The reader may question our inclusion of <u>every</u> object appearing in the meaning representation of the focus of S in one of the above focus sets, i.e. in the set of potential referents. In fact, however, it seems to be the case that <u>any</u> object (of one of the above-mentioned types) occurring in the meaning representation of the focus of S may be the referent of an object occurring later in the meaning representation of S. Consider, for example, the sentence sequences formed by taking each of the sentences of 12 below - in turn - as an immediate continuation of a text containing sentence 11 below.

11. Stan argued with his sister Fran in an attempt to convince her that she should bring Mary, whom he would like to get to know, on their planned trip to the San Diego Zoo tomorrow.

12. a. <u>He</u> was really insistent.
    b. <u>She</u> was hard to convince.
    c. <u>It</u> was useless.
    d. He thinks <u>she's</u> the prettiest one of all Fran's friends.
    e. The <u>prospect</u> really excites him.
    f. He argued that <u>it</u> wouldn't tie Mary up for more than half a day.
    g. <u>It's</u> the best one in the country, you know.

h. *She* throught *it* was a terrible idea.

i. She happened to be busy *then,* but expressed an interest in coming along another time.

Each of the *underlined* items in sentences 12a-12i references some object in sentence 11. (For the sake of clarity we present in 13 below the referents as we understand them.)

13. a. Stan
    b. Fran
    c. The attempt (to convince . . .
    d. Mary
    e. EVENT (Stan will get to know Mary)
    f. The trip
    g. The San Diego Zoo
    h. Both *she* *and* *it* are ambiuous; if *she* is taken to be "Fran," then it refers to EVENT (Fran will bring Mary ...); if *she* is taken to be "Mary"), then *it* refers to EVENT (Mary will come...)
    i. Tomorrow

The point is, of course, that any item in (the meaning representation of) a sentence, S, may be referenced by some item in (the meaning representation of) a latter sentence.

On the other side of the coin the question of identifying potential *references* is just as important as that of identifying the set of all possible referents for an object which is known to reference something. If we were concerned only with pronomial reference resolution, the problem would have a simple solution; *every pronoun is a reference*. For nominal items other than pronouns the problem is far less simple;

if a noun occurs in a text just how do we know if there is a previously occurring nominal item to which it refers? As much as we would like there to be algorithmically testable criteria, i.e. recognizable syntactic and/or semantic cues, for making the decision, there seem to be none.

Thus, the mechanism we propose considers every token appearing in the translation of a sentence as a possible reference.

At present, we hypothesize the existence of a small set, R, of relations which are sufficient to account for all instances of nominal reference. Included in this set are, at the very least, the relations <u>identity</u>, <u>member of</u>, <u>subset of</u>, and <u>part of</u>. Note that although this list of relations is quite small, it suffices to handle all the examples of reference presented thus far (i.e. those occurring in sentence sequences 1-6 and 9-12 as well as those occurring in the excerpted newpaper article above).

All of the above observations taken together lead to the following sketch of an algorithm for reference resolution.

I. As each new sentence, S, is translated into its meaning representation, the various focus sets (noun-object, event, time) are updated.

II. A set, H, is formed containing all tuples of the form (N_1, N_2, ρ) such that N_1 is a nominal item occurring in (the meaning representation of S, N_2 is an object occurring in the focus set (noun-object, event, or time) appropriate to N_1, and ρ is a member of R; H is the set of all current reference hypotheses arising from S.

III. A "judgment mechanism," discussed below, is invoked to determine the liklihoods of the correctness of the various members of H.

It is clear that following step H any further processing of reference hypotheses requires that all members of H be considered relative to one another, since the correctness or incorrectness of one may depend crucially upon that of others. In the general case not all hypotheses will turn out to be correct, and in fact some may contradict others - for instance in the case of two hypothesis-triples with identical first and second elements and different third elements.

Once it has been created, the set H is submitted to a "judgment mechanism" whose task it is to choose some of the hypotheses as valid and others as invalid. The judgement mechanism must clearly have access to the world knowledge stored in memory, and must be capable of performing inferencing of a sort which produces decisions as to the relative liklihoods of the various hypotheses.

Before giving examples of just how such a judgment mechanism might work, we should make it clear that our sense of "inferencing" is very different from Rieger's (1974). In Rieger's sense inferencing is undirected, while ours is directed toward the goal of validating hypotheses. There is, in addition, another sense in which the sort of inferencing to be done by the judgment mechanism is directed. The fact that the reasons for validating or throwing out a particular reference hypothesis (on the part of human natural language users) involve the information conveyed in local context as well as world knowledge relating to items contained in that information (and world knowledge relating to items contained in world knowledge relating to items contained in that information, etc.) constitutes a good guess as to the particular pieces of world knowledge and the rules of inference which must be involved in judging that hypothesis.

Examples of reference resolution

14 and 15 below contain components of possible meaning representations of the two sentences of sentence sequence 1 at the beginning of this paper.

14. C1: CHASED (x_1, x_2)
C2: TIME (C1, YESTERDAY)
C3: SUBSET $(x_1,$ [BOYS])
C4: SUBSET $(x_2,$ [DOGS])
C5: GREATER (SIZE (x_1), 1)
C6: GREATER (SIZE (x_2), 1)

15. C7: FALL INTO (y_1, y_2)
C8: TIME (C7, PAST)
C9: MEMBER $(y_2,$ [DITCH])
C10: MEMBER (y_1, y_3)
C11: LARGEST (y_1, y_3)

The meaning reprecentations proposed for the two sentences are $C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6$ and $C_7 \wedge C_8 \wedge C_9 \wedge C_{10} \wedge C_{11}$ respectively. Note that we are not claiming that the predicates CHASED, and FALL INTO and the constants YESTERDAY, BOY, DOG, PAST and DITCH are at the level of semantic primitives; rather, the above analyses are at just the level which we need to illustate the operation of the reference resolution mechanism. Furthermore, the symbols YESTERDAY, BOY, DOG, PAST and DITCH should be taken as pointers to the definitions of the appropriate items encoded in memory in whatever fashion. The bracketing in the notation [A], where A is a pointer to a definition, is meant to be a function which takes A into an object whose meaning is the class of items satisfying the meaning pointed to by A.

Once the translation of the first sentence of sequence 1 into its meaning representation has been completed - on the assumption that that sentence is at the beginning of the text being processed - the various focus sets will contain the following:

noun object focus: $\{x_1, x_2\}$; event focus: $\{(C_1 \wedge C_2 \wedge C_3 \wedge C_4 \wedge C_5 \wedge C_6 \quad)\}$; time focus $\{\text{YESTERDAY}\}$.

After the second sentence is translated the set, H, of reference-triple hypotheses presented to the judgment mechanism will then be the following:

$$\left\{\begin{matrix} y_1 \\ y_2 \\ y_3 \end{matrix}\right\} \left\{\begin{matrix} \text{is identical to} \\ \text{is a member of} \\ \text{is a subset of} \\ \text{is a part of} \end{matrix}\right\} \left\{\begin{matrix} x_1 \\ x_2 \end{matrix}\right\}$$

Note that no member of the event focus occurrs in H because the translation of the second sentence contains no term of the form EVENT(y); for simplicity we omit the question of time referencing.

All of the relations between y_2 and x_1 or x_2 can be ruled out on the basis of SUBSET (x_2, [DOG]) SUBSET (x_1, [BOY]), MEMBER (y_2, [DITCH]) and of the world knowledge to the effect that boys/dogs cannot be identical to, members of, subsets of or parts of ditches (of course in some weird fairy tale setting one of these might be possible and shouldn't be thrown out; but in such a case local context would inform us of the "weird" situation and the appropriate one wouldn't be thrown out.)

The hypothesis that y_1 or y_3 is a part of either x_1 or x_2 can be ruled out on the basis of SUBSET (x_1, [BOY]) and SUBSET (x_2, [DOG]), which tell us that x_1 and x_2 are sets of objects, and the world knowledge that sets don't have "parts" in the sense of the "part of" relation.

Identify between y_1 and either x_1 or x_2 can be ruled out on the basis of MEMBER (y_1, y_3) which tells us that y_1 is an individual and SUBSET (x_1, [BOY]), SUBSET (x_2, [DOG]), GREATER (SIZE (x_1), 1), and GREATER (SIZE (x_2), 1), which tell us that x_1 and x_2 are sets containing more than one object. (Remember that we're not doing axiomatic set theory in which there are no "individuals " in our sense and in which the sort of "individual" which is dealt with can be a subset of some set.)

Finally, the "member of" relation between y_3 and either x_1 or x_2 can be ruled out on the basis of MEMBER (y_1, y_3) which requires that y_3 be a set, SUBSET (x_1, [BOY]), SUBSET (x_2, [DOG]), GREATER (SIZE (x_1), 1), and GREATER (SIZE (x_2), 1), which tell us that x_1 and x_2 are sets containing more than one element each, and the fact that sets are not members of sets. (Again, we're not dealing with set theory; if in fact, we were talking about axiomatic set theory in English, then local context would contain that information, and different inferences would come into play.)

This leaves us with the following hypotheses :

$$y_3 \begin{Bmatrix} \text{is identical to} \\ \text{is a subset of} \end{Bmatrix} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix}$$

$$y_1 \quad \text{is a member of} \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix}$$

But some of these hypotheses are consistent with one another; in fact the hypotheses

$$y_3 \begin{Bmatrix} \text{is identical to} \\ \text{is a subset of} \end{Bmatrix} \quad x_i \qquad i = 1, 2$$

imply the hypotheses

$$y_1 \quad \text{is a member of} \quad x_i \qquad i = 1, 2$$

respectively because of MEMBER (y_1, y_3). At any rate, the judgment mechanism assumes at this point that either y_1 is a member of x_1 or y_1 is a member of x_2. The reader is asked to recall at this point that in presenting the *usually preferred* referents for references in sentence sequences 1-4 the claim was made that in sentence sequence 1, the *usually preferred* referent for "one" is "boys." The reason for this claim is the author's observation. that, when such a pronomial reference occurs as the surface subject of a sentence, in the absence of semantic content which discrminates among the various possible referents, most people seem to take the surface subject of the last sentence in the focus as the intended referent. The reason for this human judgment is probably that the reader/hearer takes the surface subject to be the "topic" of a sentence. If this observation is correct, the judgment mechanism should, in the current example, simply choose "one of the boys" (y_1 is a member of x_1) as the proper referent. If this observation is incorrect, the judgment mechanism should judge that there is ambiguity in the reference "one "

Sentence sequence 2 at the beginning of this paper would be handled in precisely the same manner as sentence sequence 1 up to the point at which "y_3 is a member of x_1" and "y_3 is a member of x_2" were the remaining hypotheses. The knowledge that "the dogs" referred to suffer from a strange bone-weakening disease would then cause the judgment mechanism to strengthen the likelihood that "one" refers to "dogs," thus causing "y_1 is a member of x_2" to be the preferred judgment.

Sentence sequence 16 below contains an example of EVENT reference.

16. The presidnet was shot yesterday. It caused a panic on Wall Street.

Omitting all other details of the translation into meaning representation we simply note that the primitive-level predicate into which "cause" is translated requires an object of the form EVENT (F) as its subject (i. e. if we say something like "John caused a stir" what we mean is that John did something and the event (or fact) that he did that caused a stir.) Thus, when the 2nd sentence is handled, the only possible referents for "it" will be the objects contained in the EVENT focus, namely just EVENT (the president was shot yesterday). The judgment mechanism thus must simply decide if the event (or fact) that the president was shot yesterday was likely to have caused a panic on Wall Street, a judgment which, with adequate world knowledge, should certainly be confirmed.

Sentence sequence 17 is a very similar case.

17. The president was shot yesterday. Bill told me all about it. It caused a panic on Wall Street.

In order to resolve the reference "it" in the last sentence of 17, the judgment mechanism would have to decide on the relative likelihoods of i and ii below

(i) The event (or fact) that the president was shot yesterday caused a panic on Wall Street.

(ii) The event (or fact) that Bill told me about the president being shot yesterday caused a panic on Wall Street.

Again, with the availability of reasonable world knowledge about such things as presidents, their being shot and panics, the judgment mechanism should be able to choose the proper referent for "it"

While a fully detailed specification of the judgment mechanism must await further investigation, the above examples should illustrate, at least in part, the manner in which we conceive of its operation.

Conclusions

The phenomenon with which we have been dealing is one example of what we would like to call the "creative" aspect of language use; more specifically, reference of the sort we have described - and attempted to handle - is an elliptical device necessary for effective communication; moreover, it is a device which exhibits the ability of language to "change the ground rules" in a very flexible and fluid manner in response to context.

At this point we must admit that there is an even more creative type of reference than the sort we have dealt with. 18 below is an example of this type of reference.

18. Last week I caught a cold while visiting my mother in Chicago; as usual, the chicken soup had too much pepper in it.

The interesting reference in the above example is "chicken soup." There is no item in the first sentence to which it is directly related; on the other hand, few people have any trouble resolving it by interpolating between the two sentences of example 18 the idea expressed in sentence 19 below:

19. When I get sick my mother makes me chicken soup.

If sentence 19 were available, our reference resolution mechanism would easily come up with an identity relation between the two occurrences of "chicken soup " Obviously, for our proposed mechanism to resolve this reference, some sort of inferencing must first work on the 1st sentence of 18 to produce the meaning of 19 as an inference. Thus it is clear that reference resolution and general inferencing must be interleaved.

The mechanism proposed above does not handle the entire problem. It does, however, seem to be a minimal model of reference resolution (minimal in the sense that at least this much must be going on). In addition, it provides for that control over the use of general inferencing which is required to avoid a combinatorial explosion (BOOM).

References

Clark, H.H. (1975), "Bridging", Conference on Theoretical Issues in Natural Language Processing, 10-13 June 1975, Cambridge, Mass.

Rieger, C. J. (1974), Conceptual Memory: A Theory and Computer Program for Processing the Meaning Content of Natural Language Utterences, Ph. D. Thesis, Stanford University, 1974.

Schank, R. (1972), "Conceptual Dependency: A Theory of Natural Language Understanding," Cognitive Psychology 3(4), 1972.

American Journal of Computational Linguistics Microfiche 36 : 26

HOW DOES A SYSTEM KNOW WHEN TO STOP INFERENCING?*

STAN ROSENSCHEIN**

The Moore School of Electrical Engineering
University of Pennsylvania, Philadelphia 19174

Abstract The problem of constraining the set of inferences added to a set of beliefs is considered. One method, based on finding a minimal unifying structure, is presented and discussed. The method is meant to provide internal criteria for inference cut-off.

I. Introduction

Natural language processing systems that are sensitive to the semantic and logical content of processed sentences and to the pragmatics of their use generally draw inferences. A set of formulas representing the meaning of a sentence and the 'state of belief' of the system is augmented by other related formulas (the inferences) which are retrieved and/or constructed during the processing. The problem to be investigated here is: How can this process be controlled? Can reasonable criteria be found for restraining the addition of inferences?

Top-down inferences following from the meaning of lexical items (often expressed by decomposition into primitives) are clearly bounded, if no interactions are allowed among the generated sub-formulas. This process (which we call EXPANSION) will not be discussed here. Rather, we shall be concerned with SYNTHESIS, i.e., the addition of new formulas based on the

* This work was partially suppored by NSF Grant SOC 72-05465A01.

** Author's current address: Courant Institute of Mathematical Sciences, New York University, 251 Mercer Street, New York, New York 10012.

presence of already generated lower-level formulas, which we shall call beliefs. In particular, we are concerned with inferences added because a set of beliefs is recognized as fitting a pre-defined pattern.

The question we ask is: Given an initial set of beliefs over a set of primitives, what criterion can be used to halt the process of pattern matching and associated inference addition? The major structural feature that we use to provide such a criterion is a partial order over the set of patterns.

Before pursuing this suggestion any further, let us examine some of the alternative approaches to inference and inference cut-off.

To logicians, deductive inference involves rules by which forumlas can be added to a set (which initially contains the axioms) in certain ways provided other formulas are already in the set. In general, this sort of inference is quite open-ended in that one can keep applying the rules of inference and come up with more and more formulas all of which represent 'provable' statements. The termination criterion for a particular invocation of the mechanism might be the appearance of an 'interesting' formula or the loss of interest of the inferencer, but in general the statement of the rules of inference says nothing about when to cease deriving formulas.

This paradigm from logic has been carried over into Artificial Intelligence systems, where the issue of termination is very real. The usual solution has been to invoke the inferencer under the very strict control of a supervising program which has its own goals programmed in which makes certain that appropriate criteria are applied to halt the inferencing. This is most apparent in systems written in PLANNER-like languages which has user-programmable mechanisms for controlling the proof process.

In the work of Schank and Rieger, (Sch, 75) (Ri, 74) inference has more of the flavor of free association; inferences are conceived of as expanding spheres in'inference space.' Two termination strategies are employed: (1) the discovery of a chain of inferences leading from one of the initial beliefs to another through a shared formula, or 'contact point' in inference space, and (2) the association of numerical 'strengths' to formulas so that a line of inference can be discontinued if the strength falls below a certain threshold.

Strategy (2) is somewhat unsatisfying in view of the potential arbitrariness and attendant difficulties in evaluating the role of particular numerical constants in the total behavior of a complex system. These constants, presumably, have little to do with the theoretical structure of the formal inference scheme, and as such we would call them 'external criteria.' A strategy like (1) above, on the other hand, is more 'internal' and is to be preferred.*

A goal of the present work is to formulate a reasonable internal criterion for inference cut-off which can be stated formally as part of the inference rule. To do this, we shall impose a structure on the set of patterns to be used in inferencing, and the rule for adding inferences will be formulated in terms of this structure.

The operations to be described below are explained more fully in (R,75), where a description of a computer implementation is also presented.

* See also (C,75), (W,75).

II. A Partial Order for the Pattern Set

The inference rule we are aiming for is to depend on the set of input beliefs and the set of patterns. The notion we are trying to formalize is "What does this set of beliefs suggest with respect to this set of patterns?" The particular class of inferences we are concerned with are those gotten by matching beliefs in the input set against a pattern and augmenting the beliefs with additional propositions as dictated by the pattern. We want to find the least instances of patterns which cover (include) the set of input beliefs. We will take as inferences' all propositions (an arbitrary number) which are entailed by that instance of the pattern.

Put another way, the inference operation is to 'jump to conclusions. However, it is only to jump to those conclusion required to make the resulting set an instance of the least possible pattern in the pattern set.

The key concept here is 'least' in that this is what controls how many inferences are added. What would be a suitable ordering relation for patterns and propositional beliefs? One which naturally suggests itself and which is currently under investigation relies on the relations of instantiation and superset:

(1) $p \leq q$ if q is a substitution instance of p,

and (2) $S \leq_1 S'$ if $S \subseteq S'$.

Combining these two, we say that $\{p_1,\ldots,p_n\} \leq \{q_1,\ldots,q_m\}$, where the p_i's and q_j's are propositional forms, if there is a substitution, s, for the variables of $\{p_1,\ldots,p_n\}$ such that $\{s(p_1),\ldots,s(p_n)\} \leq_1 \{q_1,\ldots,q_m\}$.

Example 1.

We adopt the notational convention of prefixing variables with '?'.

Let P = {(HAPPY ?x), (PARENT ?y ?x)}

and let Q = {(HAPPY JOHN), (GIVE MR. JONES JOHN TOY),
(PARENT MR. JONES JOHN)}.

Then P $\leq$ Q under the substitution ?x←JOHN, ?y←MR. JONES.

The 'less-than-or-equal' relation is also defined for pairs of patterns:

Example 2.

Let PAT-1 = {(P ?x ?y), (Q ?y ?z)}

and let PAT-2 = {(R ?u ?v ?w), (Q ?w ?v), (P ?u ?w)}.

Clearly, PAT-1 $\leq$ PAT-2 under the substitution ?x←?u, ?y←?w, ?z←?v.

This definition of $\leq$ is quite straightford and can be made to accomodate expressions with embeddings and predicate variables. (These are included in the implementation.)

Note that the relation '$\leq$' can be thought of an information-content comparison; if S $\leq$ S' then S' contains at least as much 'information' as S (and possibly more) either by virtue of variables having been replaced by particular constants or by additional formulas having been added to the set.

Given $\leq$ for relating pairs of belief sets, pairs of patterns, or belief-set/pattern pairs, we can now formulate the belief-set-extending function, SYNTHESIZE.

III. The Inference Operation: SYNTHESIZE

Given a set of P of patterns and an input set Bel of beliefs, SYNTHESIZE returns a set I of instantiated patterns from P such that the following three conditions all hold:

(1) (Coverage of input beliefs) For each instantiated pattern p ε I, Bel $\leq$ p;

(2) (<u>Pairwise incomparability</u>) If p,q ε I then

$p \not\leq q$ and.$q \not\leq p$;

(3) (<u>Minimality</u>) There are no other instances r of patterns in P which are not ih I and yet which are '$\leq$' to some element of I and for which Bel $\leq$ r.

The elements of I = SYNTHESIZE(Bel) represent possible minimal extensions of Bel; $\bigcap$ I represents <u>clear</u> extensions of Bel, namely the superset of Bel contained in all minimal extensions.

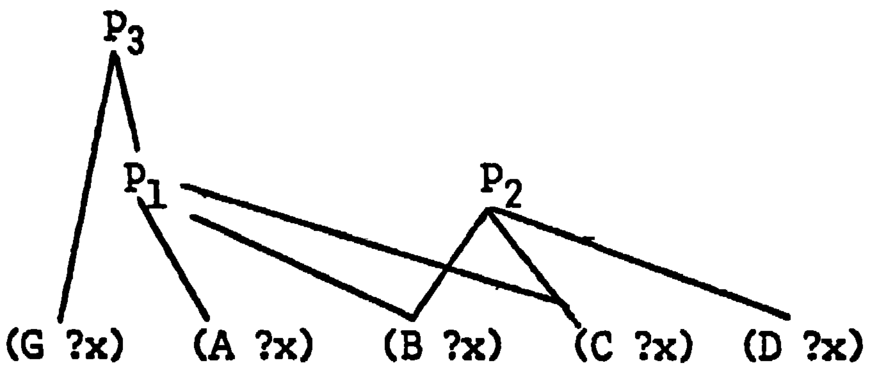
<u>Example 3</u>.

Let P = { p_1 = {(A ?x), (B ?x), (C ?x)},

p_2 = {(B ?x), (C ?x), (D ?x)},

p_3 = {(A ?x), (B ?x), (C ?x), (G ?x)} }

Represented graphically:



If input set Bel = {(A JOHN), (C JOHN)}

then SYNTHESIZE(Bel) = { {(A JOHN), (B JOHN), (C JOHN)} }.

There is only one possible minimal extension; (B JOHN) is inferred.

If input set Bel = {(B JOHN), (C JOHN)}

then SYNTHESIZE(Bel) = { {(A JOHN), (B JOHN), (C JOHN)}

{(B JOHN), (C JOHN), (D JOHN)} }.

There are two possible minimal extensions, but the set of clear extensions contains no inferences beyond the input set, Bel. (Had p_1 and p_2 shared another clause, however, an inference would have been added.)

If the input set Bel = {(G JOHN), (B JOHN)}

then SYNTHESIZE(Bel) = { {(G JOHN), (A JOHN),
(B JOHN), (C JOHN)} }.

Pattern p_3 is the least pattern which when instantiated covers the inputs, and there are two inferred propositions:
(A JOHN) and (C JOHN).

The description given here has been necessarily brief and incomplete A more formal treatment of SYNTHESIZE in terms of lattice-theoretic operations is given in (R,75) and is summarized in (JR,75). One additional technical point should be made: It often happens that for a given input set there are no single pattern instances which cover all the inputs, though patterns exist whose instances cover subsets of the inputs. In such a case we use an extended SYNTHESIZE operation which is defined in the same spirit as SYNTHESIZE. (See (R,75).)

Even without the full formal treatment, several things should now be clear. First, the actual number of inferences drawn (propositions added) for a particular input set may be small or large (depending on the inputs and the pattern set,) but it is bounded in a principled way because of the definition of SYNTHESIZE.

Second, the usual distinction between 'antecedent' and 'consequent' clauses in the pattern is not maintained; a clause in the pattern may serve as an antecedent on one occasion and a consequent on another.

Third, if 'defined' lexical items were to be associated with the patterns, noting which variables are to be bound as arguments upon instantiation, then the SYNTHESIZE function can be used to compute summarizing expressions. Thus SYNTHESIZE represents a possible formalism for lexical insertion.

IV. An Example of the Operation SYNTHESIZE

For the sake of illustration, let the primitives be:

(BENIGN ?x)

(THREATEN ?x ?y) -- ?x threatens ?y

(GIVE ?x ?ob ?y) -- ?x gives ?ob to ?y

(BELONG ?ob ?x) -- ?ob belongs to ?x

(INTEND ?x ?Q) -- ?x intends to do ?Q

(RETURN ?x ?ob ?y) -- ?x returns ?ob to ?y

(PAYS-INTEREST ?x ?y) -- ?x pays interest to ?y

(These primitives and the patterns below may appear somewhat artificial, but we have chosen a simple illustration due to the difficulties in following examples with more than a few clauses.)

Let the pattern set consist of the following four patterns:

PAT-1: ?x borrows ?ob from ?y:

{(BENIGN ?x), (BELONG ?ob ?y), (GIVE ?y ?ob ?x),
(INTEND ?x (RETURN ?x ?ob ?y))}

PAT-2: ?x takes-loan-from ?y:

{(BENIGN ?x), (BELONG ?ob ?y), (GIVE ?y ?ob ?x),
(INTEND ?x (RETURN ?x ?ob ?y)), (PAYS-INTEREST ?x ?y)}

PAT-3: ?x robs ?y:

{(NOT (BENIGN ?x)), (BELONG ?ob ?y), (THREATEN ?x ?y),
(GIVE ?y ?ob ?x), (NOT (INTEND ?x (RETURN ?x ?ob ?y)))}

PAT-4: ?x plays-practical joke on ?y:

{(BENIGN ?x), (BELONG ?ob ?y), (THREATEN ?x ?y),
(GIVE ?y ?ob ?x), (INTEND ?x (RETURN ?x ?ob ?y))}

A rough graphic representation of the set of patterns is shown in Figure 1.

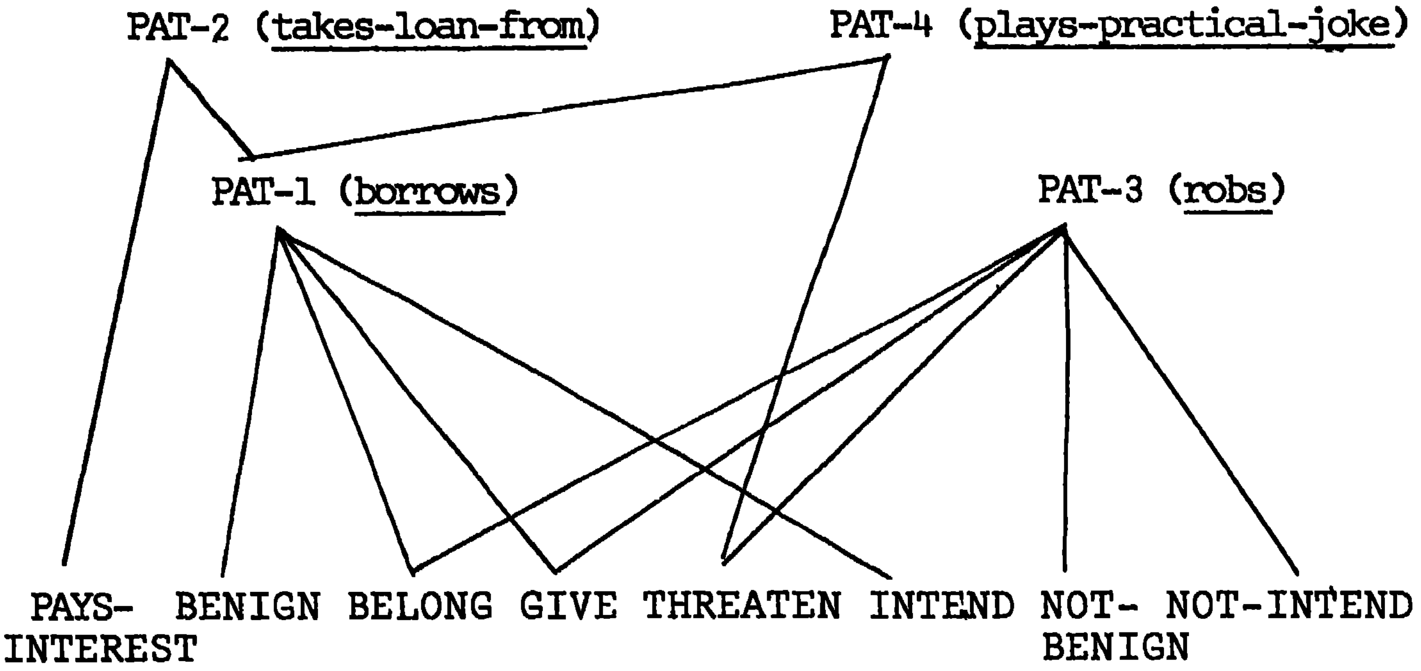


Figure 1

Now consider the following situations:

Situation 1. Input beliefs

```
Bel = {(BELONG WALLET HARRY), (GIVE HARRY WALLET MOE)}
SYNTHESIZE(Bel) =
   {{(BELONG WALLET HARRY), (BENIGN MOE), (GIVE HARRY WALLET MOE),
          (INTEND MOE (RETURN MOE WALLET HARRY))}
    {(NOT BENIGN MOE)), (BELONG WALLET HARRY), (THREATEN MOE HARRY),
          (GIVE HARRY WALLET MOE),
                         (NOT (INTEND MOE (RETURN MOE WALLET HARRY)}}
```

The minimal matched patterns are rob and borrow, adding the (conjectural) information that either Harry was threatened, on Moe intends to return the wallet.

Situation 2. Input beliefs:

```
  Bel = {(GIVE BANK 1000-DOLLARS JOHNDOE),(PAYS-INTEREST JOHNDOE BANK)}
  SYNTHESIZE(Bel) =
     {{(BENIGN JOHNDOE), (BELONG 1000-DOLLARS BANK),
           (GIVE BANK 1000-DOLLARS JOHNDOE),
           (INTEND JOHNDOE (RETURN JOHNDOE 1000-DOLLARS BANK)),
           (PAYS-INTEREST JOHNDOE BANK)}}
```

As a result of matching the loan pattern, we have added three clauses.

Situation 3. Input beliefs

```
  Bel ={(INTEND JOHNDOE (RETURN JOHNDOE 1000-DOLLARS BANK)),
         (PAYS-INTEREST JOHNDOE BANK)}
```

Here SYNTHESIZE(Bel) returns exactly the same set as was returned in Situation 2. Note, however, that the roles of

(1) (GIVE BANK 1000-DOLLARS JOHNDOE)

and (2) (INTEND JOHNDOE (RETURN JOHNDOE 1000-DOLLARS BANK))

have been reversed. In Situation 2, (1) was an input and (2) was inferred, whereas in Situation 3, (2) was input and (1) inferred. The corresponding clauses of the loan pattern were serving as antecedents on one occasion and consequents on the other. This follows naturally from the way SYNTHESIZE was defined.

In this regard the reader may notice that some input belief sets might yield 'unwarranted' or 'spurious' inferences--jumping to too many conclusions. However, the incremental addition of new patterns corrects this anomaly in a natural way: Patterns which formerly were 'least covers' may cease to be so in the extended pattern set.

V. Using Definitions to Set Up the Pattern Space

We have been particularly interested in using definitions of words to set up pattern spaces in which SYNTHESIZE could work as an inferencer and a lexical insertion technique. Special attention was payed to the 'speech act' verbs, and a brief sample list is presented below. (The symbol '?Pr' denotes a predicate variable. Also, primitive predicates are capitalized, while defined predicates are underlined.) Again, the definitions are greatly oversimplified for illustrative purposes.

```
(define tell (?x ?y ?p ?t)
        (and (BEFORE ?t0 ?t)
                (NOT (KNOW ?y ?p ?t0))
                (SAY ?x ?y ?p ?t)
                (KNOW ?y ?p ?t)
                (CAUSE (SAY ?x ?y ?p ?t)(KNOW ?y ?p ?t))))
(define request (?x ?y ?p ?t)
          (tell ?x ?y (WANT ?x ?p ?t) ?t))
(define promise (?x ?y ?Pr ?t)
          (and (FEELS-OBLIGATED ?x (?Pr ?x) ?t)
                  (tell ?x ?y (INTEND ?x (?Pr ?x) ?t) ?t)))
(define command (?x ?y ?Pr ?t)
             (and (AUTHORITY-OVER ?x ?y)
                     (request ?x ?y (?Pr ?y) ?t)))
(define implore (?x ?y ?Pr ?t)
             (and (WANTS-FAVOR-FROM ?x ?y)
                     (request ?x ?y (?Pr ?y) ?t)))
```

The expansion of these items to patterns over the primitives yields a set in which, for example, KNOW $\leq$ tell $\leq$ request $\leq$command. The input set Bel = {(BEFORE t1 t2), (SAY JAMES MASTER (INTEND JAMES (OPEN JAMES DOOR) t2) t2), (FEELS-OBLIGATED JAMES (OPEN JAMES DOOR) t2)} would be synthesized to (promise JAMES MASTER (OPEN + DOOR) t2), with added inferences (KNOW MASTER (INTEND JAMES (OPEN JAMES DOOR) t2) t2), etc., as dictated by the pattern instance of promise.

VI. Conclusion

A method has been proposed for 'free' inferencing by pattern matching in which inference cut-off can be structurally constrained: A pattern is matched if it is one of the minimal patterns whose instantiation covers the input information--even if this necessitates adding an arbitrary amount of additional information. Similarly, on the question of how many inferences to draw: Enough extra inferences are drawn to enable a coherent pattern to be matched.

The method we have proposed is general in that it makes no assumptions about the particular predicates to be used in the patterns and beliefs. (Of course, it does make assumptions about what counts as a pattern or a belief.) The inferencing could be done by a general purpose component which accepts a set of patterns as a parameter. Thus, a programmer designing a system for inference by pattern match need not devise external criteria, and certainly not criteria to be associated with every pattern. Rather the criteria are implicit in the system as a whole; any patterns which can be described in a very general pattern description language will generate its own set of internal criteria for inference cut-off.

We are continuing to investigate formalisms for structuring pattern sets in the hope of gaining further insights into this class of inferences.

References

(C,75) Clark, H.H. "Bridging," in Proceedings of the Workshop on Theoretical Issues in Natural Language Processing. Cambridge, Massachusetts., June 1975.

(JR,75) Joshi, A.K. and Rosenschein, S. "A Formalism for Relating Lexical and Pragmatic Information," in Proceedings of the Workshop on Theoretical Issues in Natural Language Processing. Cambridge, Massachusetts. June 1975.

(Ri,74) Rieger, C. Conceptual Memory. Ph.D. Thesis Stanford University. Stanford, California. 1974.

(R,75) Rosenschein, S. Structuring a Pattern Space, with Applications to Lexical Information and Event Interpretation. Doctoral Dissertation. University of Pennsylvania. Philadelphia, Pennsylvania. 1975.

(Sch,75) Schank, R., Goldman, N., Rieger, C., and Riesback, C. "Inference and Paraphrase by Computer " Journal of the A.C.M. Volume 22, No. 3. July, 1975.

(W,75) Wilks, Y. "A Preferential, Pattern-seeking Semantics for Natural Language Inference." Artificial Intelligence. Volume 6. 1975.

American Journal of Computational Linguistics *Microfiche 36 : 40*

DEVELOPING A COMPUTER SYSTEM TO HANDLE INHERENTLY VARIABLE LINGUISTIC DATA

D. BECKLES, L. CARRINGTON, AND G. WARNER IN COLLABORATION WITH
C. BORELY, H. KNIGHT, P. AQUING, AND J. MARQUEZ

Department of Mathematics and School of Education
University of the West Indies
St. Augustine, Trinidad

ABSTRACT

Linguistic communication in Trinidad and Tobago is characterised by intra- and inter-ideolectal variation in a spectrum ranging from Creole-English to Internationally Acceptable English. The tape-recorded speech of a sample of children is being analysed to determine the structure of their language, its correlation with socio-linguistic factors and their progress in the use of English. The computer system is designed to deal with manually codified data in the form of parse trees with associated grammatical and semantic information. The communication complex does not have readily identifiable norms. The analytical method and computer system effect recognition of stable sub-systems (regardless of the external criteria which determine these sub-systems), comparison of these sub-systems with English as well as state the evolution of the children's language.

Acknowledgement

The research of which this paper is a working document is partially funded by Ford Foundation Grant 690-0664D. The authors acknowledge the kind assistance of the IBM World Trade Corporation, Port of Spain, Trinidad.

Preliminary

The design and some results of the research to which the computer system relates are described by Carrington, Borely and Knight (1969, 1972, 1974 a + b). Part of the intention of the project is to describe in terms applicable to curriculum development and teacher education, the structure of the speech of school-children aged 5-11+ in Trinidad and Tobago and to compare this speech with English.

The official language and medium of instruction is English. However, the medium of daily communication ranges from a type of Creole-English to a modifed variety of Internationally Acceptable English (IAE). The term "post-creole dialect continuum" has been used by several researchers, notably Le Page (1957), De Camp (1971) and Bickerton (1973) to refer to apparently analagous situations in Jamaica and Guyana. In addition to Creole, English and variants of both, a large part of the population is exposed to a local variety of Hindi (Bhojpuri). Smaller numbers are exposed to Lesser Antillean French Creole and fewer still to Spanish.

Communication within the society is characterised by inter-ideolectal variation related to several socio-linguistic factors - ethno-linguistic background, social class, educational level, occupation, sex and age. Code-switching and intra-ideolectal variation related to the context, content and purpose of communication complicate the examination of the communication system. Since the variant levels of the complex appear to overlap they are difficult to separate into distinct sub-systems.

The Linguistic Data

The available corpus comprises 100 hours of the recorded conversation of almost 1,000 children between 5 and 11+ selected randomly from 30 schools. The data fall into two pre-determined categories: (a) free (with peer group);

(b) controlled (with investigator). Given the nature of the communication complex stated above, variation and contrast are central to the data. In addition to the usual socio-linguistic correlates of variation, these data have the possibility of containing linguistic elements which are not paralleled anywhere else in the community. These elements may occur as a result of the instability intrinsic to the performance of a vulnerable age cohort. We are not dealing with fully learned discrete languages or dialects but with partially learned systems of speech communication being used by children who, by virtue of being in school, are under pressure to abandon part of their communication repertoire in favour of another variety of speech.

Implications of the Data Type for the Analytical Procedure

English is the only code of the communication complex for which adequate grammatical descriptions are available. It is demonstrably untenable to assume that the informants are attempting to speak English at all times. They are communicating in a set of language varieties which are assumed to be rule-governed. A statement of frequency and type of deviation from English cannot therefore be an adequate analysis. The first task of the analysis must be to determine the structures, both major and minor, used by informants of various socio-linguistic descriptions.

A preliminary examination of the data shows that at the level of phrase-structure of utterances, the structures will appear to be predominantly identical with English. It is the components of the elements, their meanings and functions that will show the differences from English. Consequently, the analysis must note the levels at which derivational trees cease to be compatible with English.

In view of the variability inherent in the data, the analysis must discover the socio-linguistic correlates of the occurence of elements, as well as state co-occurence restrictions of a given element. Since it is possible that some elements may be distributed in a way that does not permit correlation with the stated socio-linguistic factors, the analysis must permit grouping of informants based on shared linguistic features for subsequent re-examination. This provision admits the possibility that sets of features may be typical of a language acquistion stage of the informants regardless of their socio-linguistic descriptions.

The Analytical Procedure

1. Each utterance is phonetically transcribed and ascribed to an informant by an identification procedure. Doubtful identity is specially coded.
2. Each utterance is rewritten in English orthography.
3. For each utterance a parse tree is constructed using the following protocol where each category described below forms the content of a node of the parse tree. The numbers are for reference and indicate the hierarchical relationship of the nodes.

Ø.Ø Utterance type	S	sentence
	SEL	elliptical S
	FRAG	fragment
	FREL	elliptical FRAG
Ø.1 Utterance complexity	SIMP	simple
	CP	compound
	CX	complex
	CPCX	compound-complex
Ø.2 Structural type	DEC	declarative
	INT	interrogative
	IMP	imperative

Ø.3 Semantic type	STMT	statement
	QU	question
	COMM	command
	RHET	rhetorical intent

Ø.4 Linear order and type of clauses occurring
e.g. MC1 + ADVC TEMP 2

Ø.5 Linear order and type of phrases occurring
(where not part of a clause)
e.g. PREP P 1 + VBL P 2

Ø.6 Dependency of clauses - dependent
embedded
co-ordinate
included

e.g. 2/1 = clause 2 is embedded in clause 1

Ø.7			ACTV	active
	AFM	affirmative	PAS	passive
	NEG	negative	EQ	equational
			STAT	stative
			LOC	locative

1.Ø surface structure of the clause/phrase occurring first.
e.g. MC1 → SUBJ + PRED* + IOBJ + DOBJ + PREP P
*PRED = predicator not predicate

1.1 detailed analysis of first occurring element of 1.Ø. e.g. SUBJ → PRMD + HDW

1.1.1 first element of subject. e.g. PRMD → [HE] PADJ, RD, MASC, SG, NOK; IAE: [HIS] etc.

2.Ø surface structure of the clause/phrase occurring second... etc to 7.Ø.

As exemplified at 1.1.1, the last node of each sub-part states the actual literal being described. The acceptability of the item as IAE is noted, OK or NOK, together with a reasonable IAE alternative. Apart from the obligatory information required by the procedure, the analyst may make additional comments which may be either in keywords or English. e.g. CMNT: probably idiosyncratic or CMNT: double NEG.

8.Ø is reserved for special idioms.

e.g. 8.Ø [SCRUNT] → scrounge for a living

9.Ø is reserved for tags.

e.g. 9.Ø TAG → [YOU HEAR]

Fig. 1 shows a sample analysis.

Figure 1

Ø652Ø72

[MY SISTER AND THEM DOES BREAK A SET OF PLATE, YES]

Ø.ØS; Ø.1 SIMP; Ø.2 DEC; Ø.3 STMT; Ø.4 MC + TAG; Ø.5 NA; Ø.6 NA; Ø.7 AFM ACTV

1.Ø MC → SUBJ + PRED + DOBJ

1.1 SUBJ → PRMD + HDW

1.1.1 PRMD → [MY] PADJ, ST, SG, OK

1.1.2 HDW → N. ASOC, ANIM, NOK: IAE: NEQV

1.1.2.1 N ASOC → NCO + ASOC

1.1.2.1.1 NCO → [SISTER] N SG, ANIM, OK

1.1.2.1.2 ASOC → [AND THEM] NOK; IAE: NEQV; VIDE 8.Ø

1.2 PRED → AUX + VT; GR @ CTN, @ PROG, PATT, NEUTTM

1.2.1 AUX → [DOES] NOK; IAE: ZERO

1.2.2 VT → [BREAK] OK TRAN

1.3 DOBJ → PRMD + HDW

1.3.1 PRMD → IND DET + N + PREP

1.3.1.1 IND DET → [A] OK

1.3.1.2 N → [SET] NCO, SG; LEX: NOK; IAE: [LOT]

1.3.1.3 PREP → [OF] OK

1.3.2 HDW → [PLATE] N PL, INAN, NOK; IAE: [PLATES]

1.3.2.1 NPL → NCO - PLZR; NOK; IAE: NCO + PLZR
1.3.2.1.1 NCO → [PLATE] @ BCL, OK
1.3.2.1.2 PLZR → ZERO, NOK; IAE: PLZR = +S, CLF
8.∅ [MY SISTER AND THEM] → [MY SISTERS]* [MY SISTER AND HER FRIENDS]
9.∅ TAG → [YES]

Glossary of keywords

ADV - adverb(ial), ANIM - animate, ASOC - associative
AUX - auxiliary, BCL - base form final cluster, C- clause
CLF - final cluster results from suffixation, CMNT - comment
CTN - completion, DET - determiner, DOBJ - direct object, GR - grammar
HDW - headword, INAN - inanimate, IND - indefinite, IOBJ - indirect object
LEX - lexical, MASC - masculine, MC - main clause, N - noun,
NCO - countable noun, NEQV - no equivalent, NEUT - netral, P - phrase
PADJ - possessive adjective, PATT - pattern, PL - plural, PLZR - pluralizer
PRED - predicator, PREP - preposition, PRMD - pre-head modifier,
PROG - progressive, RD - third person, SG - singular, SUBJ - subject,
TEMP - temporal, TM - time, TRAN - transitive, VBL - verbal,
VT - verb used transitively
* - alternative parse or meaning, @ - absence of..., [ ] enclose literals,
, - end of information set, , - minor separator.

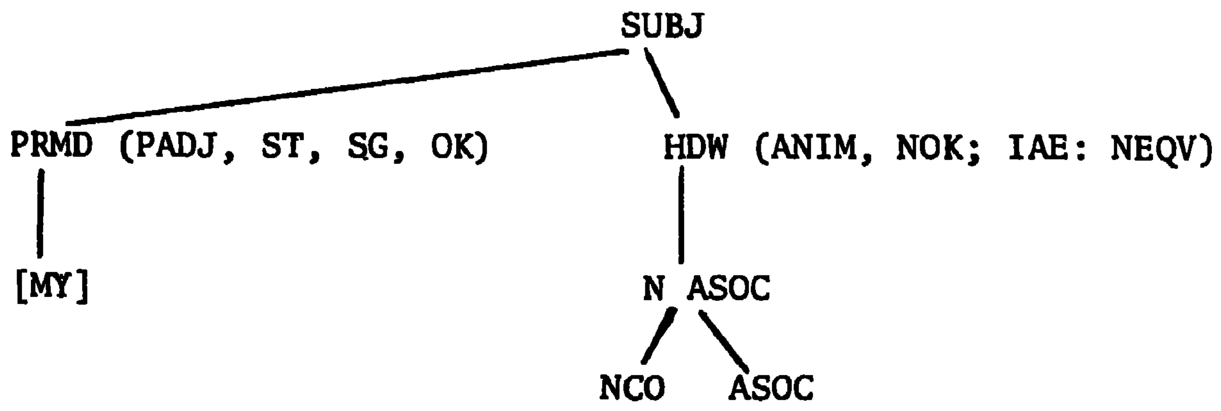
Developing the Computer System

The structure of the parse tree is, in general, quite complex and a simple *ad hoc* approach to validity checking was quickly seen to be inadequate. As a result a formal description of the tree was developed and used to construct a (partially) syntax-driven validity checking routine. The output of this routine consists of a listing of the input, with error comments where necessary, together with the internal representation of the valid trees which is written onto a file - the parse-tree file - for the subsequent analyses.

Several other files are used in addition to the parse tree file. There is the informant file which contains profiles of the informants, (e.g. age, sex, linguistic background, etc), a set of *form class* files and a set of classification files. The form class files are groupings of the various keywords which may occur in the data. Thus, for example, one form class file contains all keywords which may occur on the left-hand side of a rewrite. A classification file contains a group number for each informant; for example, one classification file contains 0 for each informant not aged 5 1 if the informant is aged 5 with a Hindi linguistic background and 2 otherwise; In any operation on the data the utterances of informants in group 0 of the relevant classification will be ignored.

Each node of a tree in the parse tree file consists of a name - in the case of a rewrite this is the left-hand side of the rewrite, otherwise it is the level number - and a set of descriptors, e.g. the grammar associated with the name. Thus, in the example of Figure 1, the lines 1.1, 1.1.1, 1.1.2, 1.1.2.1 become the sub-tree of Figure 2 where the descriptors are put in parentheses.

Figure 2



For any tree, each analysis starts at the root and many of the tasks to be described below may be regarded, in part, as a pattern matching exercise. The difficulties, and interest, arise because each node of the parse tree carries a substantial amount of information, and except for literals, only a partial matching of the nodes is usually required. In addition, some tasks require the matching of disjoint sub-trees within a given parse tree, occasionally subject to side conditions which may involve nodes not lying on the paths between the root and any of the sub-trees of interest. Apart from the pattern matching,there is the problem of classification of the occurrences of the various patterns. This is a simple tabulation complicated, in some cases, by the fact that the total number of categories is unknown.

The basic task of the system may be cast in the form: count with respect to a given classification file, and subject to stated side conditions, the occurrences of a given pattern.

Since there are only 1,000 informants and they fall into a reasonably small number of classes it is economical to pre-classify on the basis of the informant profiles rather than build the classification process into the rest of the analysis. The system is instructed to produce a classification file by a statement of the form:

CLASS = <classification file name> , (<expression list>) where <classification file name> is the name by which the file will be known, and each expression in<expression list > is a Boolean expression. For example:

CLASS = HIND1, (AGE = 5 ¢ LANG = HIND1, AGE = 5 ¢ LANG ≠ HIND1)

will produce the classification file given earlier as an example.

The side conditions refer to items in the parse trees which must occur if the tree is to be included in a given analysis. For example, if only affirmative active utterances are to be analysed the side condition Ø.7 AFM ACTV is used. The pattern to be used is stated in a manner similar to that used in specifying the input data. Thus, the pattern description

PRED → ... + AUX...; GR: @ CTN, NEUT TM, @ PROG, PATT

AUX → [DOES]

indicates that the sub-tree

```
PRED (GR: @ CTN, NEUT TM, @ PROG, PATT)
|
|
AUX
|
[DOES]
```

is of interest, subject to the convention that both the order of node descriptors (where given) and node descriptors not mentioned in the pattern are to be ignored. The occurrence of keyword FORM = < form class file name> indicates that the contents of the stated form class file are to form an additional dimension to the final tabulations. Thus the pattern

AUX → [?] FORM = OKFILE

where OKFILE contains the keywords OK and NOK and is an abbreviation for the pair of patterns.

AUX → [?] OK

AUX → [?] NOK

The symbol ? indicates that the items found there are also to add an additional dimension to the tabulations. The output of each tabulation may also be used to construct a classification file of the informants, to be used in further analyses.

CONCLUSION

In respect of performance of groups with different socio-linguistic descriptions, for purposes of this study, it is assumed that the frequency of occurrence of particular basic parse trees is a meaningful indicator of differences in speech patterns. A major difficulty is that no two trees in the study are identical but at the same time if we strip too much information from each node there are too few trees to make an analysis worthwhile, and in part, the study aims at determining the degree to which stripping of information at interior nodes is necessary if the computer is to be a useful aid.

REFERENCES

Bickerton, D. 1973 "The Nature of a Creole Continuum" *Language* 49 (3) p.640-669.

Carrington L. and Borely, C. 1969 "An Investigation into English Language Learning and Teaching Problems in Trinidad and Tobago Progress Report". U.W.I. Institute of Education, St. Augustine (mimeo).

Carrington L., Borely, C. and Knight H. 1972 *Away Robin Run*: A Critical description of the Teaching of Language Arts in the Primary Schools of Trinidad and Tobago. U.W.I. Institute of Education, St. Augustine. (mimeo).

Carrington L., Borely, C. and Knight H. 1974 "Linguistic Exposure of Trinidadian Children" *Caribbean Journal of Education* No. 1, p.12-22.

De Camp, D. 1971 "The study of pidgin and creole languages" in Hymes *Pidginization of Creolization of Languages* CUP. p.13-39.

Le Page, R.B. 1957 "General outlines of creole English dialects in the British Caribbean". Orbes 6, p.373-391.

Knight, H., Carrington L. and Borely, C. 1974 "Preliminary Comments on Language Arts Textbooks in use in the primary schools of Trinidad and Tobago". *Caribbean Journal of Education* No. 2 p.24-47.

American Journal of Computational Linguistics *Microfiche 36 : 52*

A Natural Language Processing Package

DAVID BRILL AND BEATRICE T. OSHIKA

Speech Communications Research Laboratory, Inc.
800A Miramonte Drive
Santa Barbara, California 93109

ABSTRACT

A set of SAIL programs has been implemented for analyzing large bodies of natural language data in which associations exist between strings and sets of strings. These programs include facilities for compiling information such as frequency of occurrence of strings (e.g. word frequencies) or substrings (e.g. consonant cluster frequencies), and describing relationships among strings (e.g. various phonological realizations of a word). Also, an associative data base may be interactively accessed on the basis of keys corresponding to different types of data elements, and a pattern matcher allows retrieval of incompletely specified elements. Applications of this natural language processing package include analysis of phonological variation for specifying and testing phonological rules, and comparison across languages for historical reconstruction.

I. NATURAL LANGUAGE PROCESSING PACKAGE

A. General characteristics

The natural language processing package implemented at the Speech Communications Research Laboratory (SCRL) is currently used in the analysis of associated lists of string data such as discourse transcriptions or pronouncing dictionaries. The package consists of

a) a set of "batch" programs which provide frequency and context information on the lexical and phonological forms appearing in the input; and

b) a system for interactively accessing the data on the basis of orthographic and phonological patterns.

All of the programs in this package are written in SAIL, an ALGOL-based language offering extended string and set manipulation operations and an associative data base. The programs run on a DEC PDP-10 at Carnegie-Mellon University via the Advanced Research Projects Agency (ARPA) computer network (ARPANET). The ARPANET is accessed by the ELF operating system developed by SCRL, which runs on a local PDP-11 [1].

While the processing package is applicable to various types of natural language data, it has been used most extensively at SCRL in the analysis of discourse transcriptions. The discourses consist of conversational speech gathered in interviews with adult speakers of various dialects of American English. More than twenty-five discourses, transcribed orthographically and phonologically, have been processed, yielding detailed information on over 28,000 utterances representing about 3,500 distinct lexical items. All examples in this section are taken from a typical discourse.

B. "Batch" Facility

Discourse processing usually begins with the generation of a transcription reference file in which orthographic and phonological representations are listed in discourse order, as illustrated in Figure 1.

```
897        WELL               885  //WELEHTS//
898        LET'S
899   TRY                     886  TRAY
900   CLASSIFYING             887  KLAES$CFAYIHNX
901   THEM                    888  DHAXM
902        ACCORDING=         889  //AXK$ORDIHN/TUW//
903        TO
904   THE                     890  DH$I
905   EXCUSES                 891  IHKSYUWS$CZ
```

Figure 1

In this example, the phonological realization of TRY is /trɑy/ (coded TRAY). The phonological code shown is a basic ARPA phonemic alphabet augmented by special symbols indicating some phonetic detail, such as vowel height. The realization of THE, for example, is coded DH$I, indicating that the vowel fell between /i/ and /ɪ/.

Reference numbers assigned to each utterance serve as an index to the discourse context in which utterances occur, and are used to interpret the output of other programs in the package. Separate reference number sequences are provided for the orthographic and phonological forms in the reference files, since there may not be a one-to-one correspondence between these forms, as in the case of phonological merging which obscures word boundaries. In Figure 1, for example, the two orthographic items WELL and LET'S are realized as a single phonological item /wlɛts/ (coded WELEHTS)

The core of the "batch" processing facility is a set of three programs: PROCON, ENVIRN and CLUSTR. PROCON provides frequency and context information on the lexical level, while the other two provide similar information on the phonological level.

PROCON output contains an alphabetically sorted list of the utterance types occurring in the input discourse transcription file as illustrated in Figure 2. Frequency of occurrence of each type is given, along with the various phonological realizations. For each phonological realization, frequency count and reference numbers are provided.

```
8    HAVE    3    AXV         11,337,703
             3    HHAEV       354,828,1397
             1    HHAXV       710
             1    HH$GV       1067
```

Figure 2

In Figure 2, for example, HAVE occurred eight times, and was pronounced AXV (/əv/) three times and HHAEV (/hæv/) three times. Using the reference numbers associated with these pronunciations, it is possible to establish the discourse context.

One would find that the three AXV pronunciations (i.e. utterances 11, 337 and 703) all involved the auxiliary construction in "...may have felt...seemed to have been ..which have since been..."

ENVIRN tallies occurrences of phonological segments and environments in the discourse transcriptions. The output of this program lists frequencies of all phonemes appearing in the input file, as illustrated in Figure 3.

```
          Q  30


                    D--EN       1     486
                    EH--EN      8     189, 200, 223, 226, 233,
                                      248, 368
                    ER--EN      2     1427, 1444
                    EY/--Y      1     1361
                    IH--/DH     1     134
                    IH--/IH     1     1416
                    IH--/S      1     120
                    K/--/DH     1     1178
```

Figure 3

Glottal stop, coded Q, occurred a total of thirty times in the discourse. The immediate environments of Q are listed alphabetically by left context, with word boundaries indicated by slash /, and a frequency count and reference numbers are given for each environment. For example, Q appeared eight times in the context EH--EN (/ε--n/), and a check of the reference list shows that all these occurrences were in the word sentence(s).

ENVIRN output also provides a frequency ordered list of phonemes, with frequency totals broken down according to occurrence in word initial, medial and final position.

CLUSTR, the third of the "batch" programs,is used in the analysis of phoneme cluster distribution in the discourse data. All clusters are indexed by each of their component phonemes, so that the cluster NDZ (/ndz/) which is listed under D in Figure 4 also appears under N and Z in the full output.

```
D    70
          B D           1   169
          D EN          2   71, 103
          D EN T        2   593, 1127
          D EN T S      1   699
          D Q EN T S    1   486
          D V           2   1417, 1445
          D Z           5   278, 284, 837, 1341, 1350
          L D          12   38, 385, 616, 712, 923, 1248,
                            1465, 1474, 1478, 1480, 1494,
                            1512
          M D           1   330
          W D          32   35, 118, 186, 227, 328, 400,
                            419, 550, 608, 608, 627, 631,
                            653, 670, 682, 704, 717, 730,
                            730, 745, 853, 933, 1039, 1194,
                            1199, 1201, 1228, 1233, 1253,
                            1320, 1372, 1425
          N D Z         1   1429
          R D           5   630, 889, 1277, 1303, 1314
          V D           3   187, 750, 765
          Z D           2   106, 451
```

Figure 4

Separate output may be generated for clusters occurring within words or across word boundaries. Currently, consonant and vowel clusters are tallied, but the program can be easily modified to handle sequences of phonemes belonging to arbitrary user-defined classes (e.g. voiced sounds, nasals, unvoiced stops, etc.).

For each phoneme belonging to a selected class, CLUSTR provides a count of the number of times that the phoneme appears in clusters, an alphabetically sorted list of those clusters, and a frequency count and reference numbers for each cluster. Figure 4, a sample of CLUSTR output for within-word consonant clusters, shows that D appeared in clusters a total of 70 times, with 32 of these being ND clusters. Reference numbers may be used to establish the discourse context of any cluster. For example, the cluster D Q EN T S (/dʔnts/) appears in utterance 486 which is the word students. Like ENVIRN, CLUSTR provides a frequency ordered list of cluster types in addition to the alphabetic list.

C. Interactive Retrieval Facility

The set of "batch" programs is complemented by a language data retrieval system which allows the user to interactively retrieve data items conforming to various orthographic, phonological and syntactic patterns.

Linguistic data is internally stored in the system as a network of associations between items of various types. These associations are implemented in SAIL as LEAP triples [2] and the element types entering into these associations vary according to the particular application. For example, in analysis of the discourse data described above, triples contain orthographic, phonological and syntactic elements. For study of phonetic-to-phonemic mapping, triples might be orthographic, phonemic and phonetic elements. In comparative linguistic research, triples might consist of an orthographic element and two phonological elements corresponding to two languages or dialects

Data can be accessed on the basis of patterns directed to any one (or any combination) of these elements. For example, if the data base contains associations between orthographic, phonological and syntactic elements, then the query

P/ O: THE

retrieves the phonological items associated with the spelling THE, and might return DHAX (/ðə/) and DHIY (/ði/). The query

O/ P: TUW

would return the orthographic items pronounced TUW (/tu/), e.g *two*, *too*, *to*.

Patterns such as THE and TUW completely specify the element to which they are directed, but various special forms allow partial specifications to be expressed also. The symbol $ matches any single segment (in a phonological pattern) or character (in an orthographic pattern), and the symbol = matches any number, including zero, of contiguous segments (or characters). Thus, if N is the syntactic code for Noun, the query

O/ P: $$, S: N, O: D=

searches for all two-phoneme nouns which begin with the letter D, and might return *dye*, *day*, *doe*, *dough*.

Each phonological element is defined in terms of a set of features such as UV (unvoiced) and ST (stop), and these features may be used to specify segments in phonological patterns. To search for phonological realizations containing /i/ between unvoiced stops, one could use the query

P/ P: =⟨UV + ST⟩IY⟨UV + ST⟩=

to find /kip/ (*keep*), /pikɪŋ/ (*peeking*), and /rɪpit d/ (*repeated*)

Boolean operators are also available for specifying pattern segments. For example, the query

O/ Ò: (C OR K)=, P: (NOT K)=

returns orthographic items which begin with C or K and are not pronounced with initial /k/, e.g. *cite*, *change*, *know*.

Several capabilities lacking in the current interactive system will be available in the near future. The user will be able to (1) specify optional segments and sequences of segments in phonological patterns; (2) create and name sets containing items of interest, e.g. monosyllabic function words, and use set operations such as union and intersection; (3) interactively modify feature definitions of phonological symbols; (4) retrieve several elements, e.g. orthographic and phonological forms, simultaneously; (5) display the discourse context of any given item, and (6) write retrieval queries and responses to a file for subsequent analysis.

II. APPLICATIONS

The processing package can be used in the analysis of various kihds of natural language data, as illustrated in the following examples.

A. Phonological variation

The programs can be used to efficiently index and sort natural language data so that systematic phonological variation can be easily examined. For example, inspection of a PROCON output for a ten minute interview consisting of over 2,000 utterance tokens yields general observations such as

-- final /t/ alternates with final glottal stop /ʔ/ under certain conditions;

-- alveolar flapping occurs under several stress conditions which appear to be related to noun affixes.

These preliminary observations can be systematically investigated using the interactive query system.

The data base can be queried for all phonological realizations ending in T (/t/) or Q (/ʔ/), and the corresponding orthographic entries, using the queries

P/ P: =(T OR Q)   and   O/ P: =(T OR Q)

The resulting list might include

art	/ɑrt/	limit	/lɪmɪt/
but	/bət/		/lɪmɪʔ/
	/bəʔ/	raft	/ræft/
can't	/kænt/	that	/ðɛt/
	/kᵊnʔ/		/ðɛʔ/
fished	/fɪʃt/	want	/wɑnt/
it	/ɪt/ /ɪʔ/		/wɑnʔ/

That is, final /t/ appears to vary with final /ʔ/ following vowels and following nasals, but not elsewhere. This hypothesis, represented as a context-sensitive phonological rule, could then be tested against additional data using any of several computer rule testers [3-5].

Forthcoming modifications will allow queries with set operations, such that the intersection of orthographic entries having final /t/ alternating with /ʔ/ can be requested directly by the query

O/ P: =T ∩ O/ P: =Q .

That is, only entries with /t/ and /ʔ/ alternation would be retrieved, and the entries *art*, *fished* and *raft* would not be returned.

In order to determine the conditions under which alveolar flapping occurs, the queries

O/ P: =DX= and P/ P: =DX=

can be used to retrieve phonological items which contain DX (/ɾ/) and corresponding orthographic items. Such a list might include

ability	/əbílɪɾi/
city	/síɾi/
facility	/fəsílɪɾi/
letter	/léɾɚ/
petty	/péɾi/
responsibility	/rɛspɑnsɪbílɪɾi/
writing	/rɑ́yɾɪŋ/

Flapping occurs in a descending stress pattern, e.g. city letter, petty, writing in which a stressed vowel precedes the flap and an unstressed vowel follows. In addition, the flap appears to occur between unstressed vowels when the sequence represents the noun affix -ity, as in ability. To check this, the query

P/ O: =ITY, S: N

could be used to retrieve all nouns ending in -ity, and the subset involving affixed forms (i.e. excluding city, pity) could be examined for occurrences of flapping.

B. Word Error Recognition testing

The interactive facility can be used to examine the kinds of word recognition errors which might occur in a speech understanding system due to indeterminacies in segment labelling. If a string is completely specified as /likɪŋ/(coded LIYKIHNX), then it matches a single word, leaking. However, if labelling is less precise, then alternative (and incorrect) word matches might occur. Using the interactive retrieval system, alternative labels and resulting word matches can be examined for any given lexicon.

In the example above, the labelled string might be

L ⟨VOC HIGH ANT⟩ K IH NX

with the stressed vowel represented as a set of features: vocalic, high, anterior. Resulting word matches might include leaking and licking.

If the initial consonant is also specified as a set of features (consonant, sonorant, continuant), as in the string·

⟨CON SON CONT⟩ ⟨VOC HIGH ANT⟩ K IH NX

then the resulting word matches might be leaking, licking, reeking. If the K is specified less precisely as a voiceless stop, word matches might include leaking, licking, reeking, leaping, ripping.

The interactive facility allows the system designer to easily determine the nature of possible incorrect matches due to phonological indeterminacy, especially as the size of the lexison increases.

C. Comparative Linguistic Relationships

If the data base is represented as an orthographic list with two associated phonological lists representing two languages or dialects, the interactive system can be used to discover systematic sound correspondences, and to aid in the study of dialect relationships and historical reconstruction.

A sample'data base might be:

Gloss	Language A	Language B
a fish	plaa	pa
to have	mii	mia
no, not	plaaw	paw
brother	phii	fia
bamboo	phaay	fay

The query

B/ A: PL=

would retrieve those items in language B which correspond to items in language A with initial /pl-/ clusters, e.g. pa and paw, indicating that consonant cluster simplification may have occurred in language B. The query

B/ A: =IYIY

would retrieve those items in language B which correspond to items in language A with final /-ii/, e.g. the diphthongized mia and fia.

A large data base could be accessed in this way to discover systematic correspondences between languages A and B, such as the correspondences /pl-/:/p-/, /m/:/m/, /ph-/:/f/, /-ii/:/-ia/, /-aa/:/-a/, etc.

The flexibility of the interactive system, combined with the linguistic intuition of the user, can be used.to specify and retrieve any set of correspondences, without the need to format the data according to initial consonants or clusters, vowel nuclei, finals, etc. Information such as tonal contours and stress can also be represented and accessed.

REFERENCES

[1] Retz, D. L., J. R. Miller, J. L. McClurg, B. W. Schafer, Elf Kernel Programmer's Guide, Speech Communications Research Laboratory, Santa Barbara, California. April, 1975.

[2] Feldman, J. A. and P. Rovner, "An ALGOL-based Associative Language," Comm. ACM, Volume 12, August, 1969, 439-449.

[3] Barnett, J. A., A Phonological Rules System, TM-5478/000/00, System Development Corporation, Santa Monica, California, 1975.

[4] Bobrow, D. G. and J. B. Fraser, "A Phonological Rule Tester," Comm. ACM, Volume 11, November, 1968, 766-772.

[5] Friedman, J. and Y. C. Morin, Phonological Grammar Tester:

Description, Natural Language Studies No. 9, Phonetics Laboratory, The University of Michigan, 1971.

ACKNOWLEDGEMENT

This research was supported in part by the Advanced Research Projects Agency of the Department of Defense through Contract N00014-73-C-0221 administered by the Office of Naval Research Information Systems Program.

ON THE ROLE OF WORDS AND PHRASES IN AUTOMATIC TEXT ANALYSIS

G SALTON
Cornell University

Automatic indexing normally consists in assigning to documents either single terms, or more specific entities such as phrases, or more general entities such as term classes. Discrimination value analysis assigns an appropriate role in the indexing operation to the single terms, term phrases, and thesaurus categories. To enhance precision it is useful to form phrases from high-frequency single term components. To improve recall, low-frequency terms should be grouped into affinity classes, assigned as content identifiers instead of the single terms.

Collections in different subject areas are used in experiments to characterize the type of phrase and word class most effective for content representation.

The following typical conclusions can be reached:

a) the addition of phrases improves performance considerably;

b) use of phrases is better with corresponding deletion of single terms in practically all cases;

c) the use of both high-frequency and medium-frequency phrases is generally more effective than the use of either phrase-type alone;

d) the most effective thesaurus categories are those which include a large number of low-frequency terms;

e) the least effective classes either consist of only one or two terms, or else they include terms with unequal frequency characteristics permitting the high-frequency terms to overcome the others.

The discrimination value theory is developed and appropriate experimental output is supplied.

American Journal of Computational Linguistics *Microfiche 36 : 68*

GRAMMATICAL COMPRESSION IN NOTES AND RECORDS: ANALYSIS AND COMPUTATION

BARBARA B. ANDERSON

Department of Anthropology
University of New Brunswick

IRWIN D. J. BROSS

Roswell Park Memorial Institute
Buffalo, New York

NAOMI SAGER

Linguistic String Project
New York University
2 Washington Square Village, 2B
New York, New York 10012

ABSTRACT

Linguistic mechanisms of compression are used when making notes within a context where the objects and meanings are known. Mechanisms of compression in medical records for a collaborative study of breast cancer are described. The syntactic devices were mainly deletion of words having a special status in the grammar of the whole language and deletion in particular positions of words having a special status in the sublanguage. The deleted forms are described and sublanguage word classes defined. A subcorpus of the medical records was parsed by an existing computer parsing system; a component covering the deletion-forms was added to the grammar. Modifications to the computer grammar are discussed and the parsing results are summarized.

Introduction

All languages have mechanisms of compression. Sentences may be embedded within other sentences by means of nominalization and complementation. Various grammatical transformations involve deletion of certain parts of the sentence.

In medical records, we find entries such as no evidence of metastases, which may be said to be derived from something like There is no evidence of metastases. Such incomplete sentences are not common in the spoken language of the medical records (i.e. dictated reports). However when physicians themselves are required to write material for records, compression mechanisms are commonly used.

Although this paper will deal with a specific corpus, similar devices would often be used for compression in other situations where there is pressure to write as little as possible. Legal, educational, and scientific records where informal notes are kept would be other examples of this class of situations.

The original motivation for this study was to develop effective methods for storing the information in a medical record and to be able to retrieve this information for purposes of research, medical care, or administration. From previous research, the feasibility of verbatim input of dictated narrative has been established. Computerized extraction of the information has been shown to be feasible in a test system ACORN (Automated Coding of Report Narrative). This system has been described in detail in a series of previous papers.[1,2,3]

[1] I.D.J. Bross et al. "Information in Natural Languages: A New Approach". Journal of the American Medical Association, Vol. 207, No. 11, 1969, pp. 2080-2084.

[2] I.D.J. Bross et al. "Feasibility of Automated Information Systems in the User's Natural Language". American Scientist, Vol. 57, No. 2, 1969, pp. 193-205.

[3] P.A. Shapiro and D.F. Stermole. "ACORN (Automated Coding of Report Narrative): An Automated Natural-Language Question-Answering System for Surgical Reports". Computers and Automation, Vol. 20, No. 2, 1971.

For a highly structured medical record where the entries are single words or very restricted sentences, the feasibility of computer-assisted editing and coding has also been established. A procedure for typing in the entries verbatim in a medical record, called 'TICES' (Type-In Coding and Editing System) has been reported elsewhere.[4] However, the third, intermediate class of material cannot be handled by ACORN or by TICES. Therefore, a linguistic analysis of this type of material has been undertaken with the ultimate objective of setting up a comprehensive computer system that can handle almost everything in the medical records.

In the earlier efforts to develop natural language technology, the work was facilitated by the fact that the documents involved were strictly for the transmission of factual information.[5] Such documents are regarded as important both by the persons who are filling them out and by the persons who read them. In this no-nonsense situation where the record may be critically reviewed by the peers of the person who is reporting the information, unambiguous and informative transmission of information is a critical need. Some of the simplicities in the present analysis may be peculiar to this type of situation.

The existence of a subculture with shared training, objectives, and experience may facilitate the note-taking process in somewhat the same way that a person taking notes for himself can somehow be more concise without ambiguity. However, many other note-taking situations would involve a subculture, though not necessarily a medical one, and the findings here might be expected to have some general applicability.

[4] I.D.J. Bross et al. "Unobtrusive Biomedical Data-Input Systems". Bio-Medical Computing, No. 4, 1973, pp. 219-228.

[5] I.D.J. Bross, P.A. Shapiro and B.B. Anderson. "How Information Is Carried in Scientific Sublanguages". Science, Vol. 176, No. 4041, 1972, pp. 1303-1307.

Source of Material

The medical notes discussed here are from the records of the Surgical Adjuvant Breast Project, a nationwide collaborative study involving 36 medical institutions. The records were filled out by medical and paramedical personnel at the participating institutions and centralized at Roswell Park Memorial Institute in a statistical unit under the direction of Dr. Nelson Slack. A sample of approximately 50 was taken from the 2734 case histories of patients in the program and is being used in the linguistic analysis. Each case history ordinarily consists of 3-6 pages of detailed information on the patient's initial status, treatment, pathology report, medical problems, and subsequent fate. When the structured information in the record was excluded, each case history had between 6 and 26 notated items, each item consisting of 1 to 5 partial sentences. While this material is specialized to the purposes of the collaborative study, this type of information is fairly typical of what is found in the usual hospital record.

The notes were typed verbatim using an IBM Mag Card Communicator so as to obtain simultaneously a typed paper document and a record in computer-usable form. This device is used in the data-input system of TICES, an existing system for handling completely structured records. It would presumably be used in any extension of TICES which would handle medical notes. In this analysis the computer was used to reorganize the material in a form more convenient for manual analysis by the linguist.

Anderson analyzed the linguistic structure of the entries in a sample of the medical records involving radiation findings. A discussion of this analysis will take up the next part of the paper. Sager and associates used some of the findings from this study to develop methods for processing these same medical records by computer, adapting a program and grammar which had been developed for parsing science articles. This project will be discussed in the final part of the paper.

Linguistic Characteristics of Medical Notes

Many of the entries on the medical records are in the form of notes which are neither complete sentences nor single word entries, but linguistic strings of an intermediate type, which we will hereafter call fragments. Fragments are a compressed type of linguistic material resulting from various transformations which have the effect of making linguistic strings shorter by reducing or deleting material. The writer of these stretches of material must make his entries brief, in order to save time and effort, but also make them informative and unambiguous. For this reason the deleted material has to be easily recoverable, or in other words it must not contain much information. An analysis of the fragments shows that deletion is mainly of a small class of sentence parts: (1) tense and the verb be (t be); (2) subject, tense and the verb be; (3) the subject; and (4) subject, tense, and verb (V) other than be.

A second characteristic of fragments which makes deleted material recoverable is that both the deleted material and the remainders consist of words in easily defined subclasses, based on both distributional and semantic criteria. These subclasses are easily defined because of the nature of the sublanguage; in general the vocabulary is limited and each word has a limited semantic range. The question on a form which is being answered can also be used as a basis for restoring deleted material.

One of the most commonly deleted items in the medical records is t be (1 and 2). Tense is perhaps the most important information be gives. The deletion of tense in the medical records causes no ambiguity because usually the physician describes the situation at the time of filling out the report. Otherwise he gives the time in a time phrase: x-rays on November 2.

Fragment Types

In Table 1 we list the fragment types, giving an example of each, but not with all occurring word subclasses. The types will first be given according to what material is deleted and then will be further subclassed according to the two highest nodes of the tree structure of the remainder. The material in brackets is the word subclasses which are assumed to have been deleted.

TABLE 1. FRAGMENT TYPES

Material Deleted	Structure of Fragment	Example
1. t be by N-physician	N Ven	no metastatic lesions [were] detected [by physician]
	N Adj	chest films [were] normal
	N P N	patient [was] without cough
	N to V	this form [is] to be used . . .
	N Ving	wound [is] healing well
2. Subject t be	Ven	[N-disease was] aspirated once
	Adj	[N-patient is] dead
	to be Ven	[N-patient is] to be seen by gynecologist
	Ving	[N-patient is] doing well
3. Subject	t V Object	
3a. N-physician Subject		[I] found osteochondritis in rib (5th right)
3b. N-patient Subject		[N-patient] had period one week ago
3c. N-disease Subject (rare)		[N-disease] invades skin [N-disease] seems minor
4. Subject t v		
4a. N-physician t V-discover	Object	[I V-discovered] no bony metastases
4b. N-physician t V-do	Object	[N-physician did] excision of (r) 5th costal cartilage
4c. N-patient t have	Object	[N-patient has] no bone pain

Word Subclasses

The word subclasses should have three characteristics: (1) they should enable deleted material to be recovered, (2) they should make it possible to extract and store informational units such as those in ACORN[6] and (3) they should be defined so that a linguistically unsophisticated person can easily put words into their subclasses.

The word subclasses are based on both semantic and distributional criteria. To a large extent nouns can conveniently be subclassed on a semantic basis and verbs can be subclassed on a distributional basis, according to the subclasses of nouns which they take as subject and object. Due to the nature of the sublanguage there is relatively little overlap (e.g., a given verb is likely to take only one noun subclass as subject) compared to what we would find in the language as a whole.

Two important subclasses of human nouns used in the medical records are N-physician and N-patient. Each has only a few members, but is important because many verbs characteristically take it as subject or object, and also because both, but particularly N-physician, are usually deleted. It is on the basis of the verbs which characteristically take them as subject or object that they can usually be recovered without ambiguity.

Other noun subclasses concern more directly the subject matter of the reports, the concrete objects with which the physician is dealing. Unlike N-physician and N-patient, these classes usually have many members and they are seldom deleted. As with N-physician and N-patient, certain verb subclasses characteristically take them as subject or object.

Table 2 gives some of the word subclasses with examples of each.

[6] Bross et al. "Information in Natural Languages: A New Approach," 1969.

TABLE 2. SOME WORD SUBCLASSES

1.	N-bodypart	abdomen, axilla, bone, breast, cervix, pelvis
2.	N-change	change, elevation, enlargement, gain, increase
3.	N-dimension	pressure, rate, rhythm, size, weight
4.	N-disease	carcinoma, cough, disease, edema, fibrosis
5.	N-exam	biopsy, exam, film, mamogram, scan, x-ray
6.	N-location	area, field, floor, lobe, neck, part, region
7.	N-patient	she, her, patient, lady, woman
8.	N-physician	doctor, he, him, his, I, M.D., radiologist
9.	N-therapy	drug, insulin, medication, medicine, radiation
10.	N-time	date, month, time, visit, winter, year
11.	V-be-equivalent	appear, feel, indicate, remain, represent, seem
12.	V-change	alter, clear, change, enlarge, heal, progress
13.	V-discover	detect, find, identify, note, observe, see
14.	V-patient-object	admit, give, leave, place, readmit, see, transfer, treat
15.	V-patient-subject	complain, come, cooperate, enter, feel, gain, go, have, refuse, show, suffer, take
16.	V-physician-subject	feel, have, place, tell, transfer, treat, see
17.	V-show	show, demonstrate, indicate, reveal, suggest
18.	Adj-bodypart	axillary, bony, clavicular, lumbar, pelvic
19.	Adj-changed	elevated, enlarged, healed, stable, unchanged.
20.	Adj-degree	considerable, extensive, intermittent, little
21.	Adj-discover	absent, evident, known, possible, present
22.	Adj-disease quality	active, bad, benign, degenerative, firm, hard, malignant, metastatic, nodular
23.	Adj-location	adjoining, distal, dorsal, frontal, left
24.	Adj-negative	clear, free, healthy, negative, normal

Computer Parsing of Medical Records[7]

To test the linguistic analysis, a subset of the manually analyzed corpus of medical records was parsed by computer, using the NYU Linguistic String Parser.[8]

[7] I am grateful to Cynthia Insolio and Lynette Hirschman for their help in processing these data. (N.S)

[8] R. Grishman, N. Sager, C. Raze, and B. Bookchin, "The Linguistic String Parser". Proceedings of the NCC, AFIPS Press, Montvale, N. J., 1973.

The LSP grammar of English is based on the same linguistic principles as the ACORN grammar. Hence it could also serve to test the feasibility of adding a note-handling capability to the ACORN-TICES system. The LSP sy[illegible] which was designed for text-processing, was adapted to the parsing of medical records by deleting portions of the grammar which are not required for this type of material and adding a section covering sentence fragments. These changes are described below, followed by the parsing results.

The corpus which was parsed consisted of 12 sections of the Radiation Findings extracted in their order of appearance from the medical records. These sections contained 245 sentences or sentence fragments (word sequences ending in a period). Of these, 37 were complete English sentences and 205 were fragments; 3 were combinations of both types. 21 entries were identical to others in the corpus, accounting in all for 139 of the sentences or sentence fragments. Of the complete sentences, some were quite long, e.g., *Reexamination shows some scarring and thickening over the right apex which is perhaps slightly more evident than it was before, but nothing is seen that is typical of tumor involvement*. Typical shorter sentences are *Chest films on 10-25-68 and 12-14-68 do not show any essential changes since last reports, Liver scan 1-29-69 was normal*. Fragments were, as predicted, of the types listed in Table 1, above, though not all types were represented in the parsed corpus.

Table 3 shows the new definitions or redefinitions which were added to the LSP grammar to cover fragments. These definitions are written in Backus-Naur Form (BNF), as are all the ca. 180 definitions which comprise the context-free part of the LSP English grammar. The BNF definitions are used by the parser to construct a tree representing the structure of the input sentence.

In addition to BNF definitions, the grammar contains restrictions, which test the sentence trees for grammatical and selectional well-formedness.[9] The

[9] For more explanation of the LSP system and grammar, see N. Sager and

TABLE 3. DEFINITIONS ADDED TO THE LSP GRAMMAR
TO COVER SENTENCE FRAGMENTS

```
 1.  <SENTENCE>      ::= <TEXTLET>.
 2.  <TEXTLET>       ::= <OLD-SENTENCE><MORESENT>.
 3.  <OLD-SENTENCE>  ::= <INTRODUCER><CENTER><ENDMARK>.
 4.  <MORESENT>      ::= NULL/<TEXTLET>.
 5.  <INTRODUCER>    ::= NULL.
 6.  <CENTER>        ::= <ASSERTION>/<FRAGMENT>/<IMPERATIVE>.
 7.  <FRAGMENT>      ::= <SA>        (<SOBJBESHOW>/<ASTG><SA>/<NSTG><SA>/
                         <VENPASS>/<NSTG>(<ASSERTION>/<SOBJBESHOW>)).
 8.  <SOBJBESHOW>    ::= <SUBJECT><BE-OR-SHOW><OBJBE><SA>.
 9.  <BE-OR-SHOW>    ::= ↓--↓/NULL.
10.  <ENDMARK>       ::= ↓.↓/↓,↓/↓;↓/↓--↓.
```

starting, or root, definition of the grammar is SENTENCE, so this is the first definition seen in Table 3. In the case of medical records, the unit may be longer than one sentence, but we have retained the root-word SENTENCE and defined SENTENCE in this case to be a TEXTLET (definition 2), which consists of a sentence (called OLD-SENTENCE, definition 3) optionally followed by more sentences (MORESENT, definition 4). The definition of OLD-SENTENCE has the same three elements (INTRODUCER, CENTER, ENDMARK) that the definition of SENTENCE does in the LSP grammar; however, in this case, the INTRODUCER (definition 5) is NULL; the CENTER (definition 6) contains an option FRAGMENT in addition to the options ASSERTION and IMPERATIVE defined in the English grammar (other options of CENTER, e.g. QUESTION, have been deleted); and the ENDMARK (definition 10) contains unconventional punctuation, such as dashes and comma, in addition to the period and semicolon. Since our main interest here is in FRAGMENT (definition 7), we will elaborate on this definition.

R. Grishman, "The Restriction Language for Computer Grammars of Natural Language' Commun. of the ACM, 18, 390-400, 1975, and the references cited there.

In defining FRAGMENT, we have used parts of the grammar which were defined independently of the fragment problem. That this is possible is in itself a partial verification of the conclusion from manual analysis that only limited, grammatically specifiable, deletion-forms occur in the fragments seen in notes and records. For example, the dropping of the verb *be* (type 1 of Table 1) can occur in normal English when a sentence containing the verb *be* occurs as the object of a verb like *find*, e.g. *We found the chest clear to percussion and auscultation*. In the LSP grammar there is an object string defined for such occurrences; it is called SOBJBE (Subject + Object of *be*). This same string can then be made an option of CENTER to analyze fragments having the same form e.g. *Chest clear to percussion and auscultation*.

In detail, the definition of FRAGMENT begins with the element SA (Sentence Adjunct). The definition of SA (not shown here) contains 16 options covering all types of sentence modifiers. In this material the most frequent SA is a time expression, usually a date (called PDATE, for optional Preposition + *date*) or *this examination*, *this visit*. Following SA in the definition FRAGMENT are the options proper, naming definitions already in the LSP grammar. The first option SOBJBESHOW (Subject + Obj ect of *be* or *show*), corresponds to the second and third structures of type 1 and also occurrences like *Chest film no change*, which is an expansion of SOBJBE, discussed above. This option covers deletions of the two most common verbs in this material, *be* and *show*. The place of *be* or *show* (definition 8) in a fragment is either empty or is filled by a dash.

The second and fourth options, ASTG and VENPASS, in FRAGMENT correspond to structures of type 2 in Table 1 (e.g., *Negative*, *felt to be a benign lesion*), where the subject, tense and verb *be* have been dropped. In the LSP grammar, ASTG (Adjective string) is an option of OBJBE, and VENPASS (V-en passive string) is also permitted after *be*, and in other places. The third option, NSTG (Noun string), is an object of *show*, e.g., *Mild degenerative changes* (from, *X-rays show mild degenerative changes*). It also covers occurrences of the first structure of type 1 (e.g. *No X-rays taken*) where for regularity with more complete entries the passive verb (*taken*) is seen as a right adjunct of the noun. The last option, consisting of NSTG followed by either ASSERTION or SOBJBESHOW, covers such occurrences as *PA and lateral chest 11-5-71 reexamination shows some scarring and thickening over the right apex*. where a noun phrase (*PA and lateral chest 11-5-71*) precedes an assertion about that noun phrase.

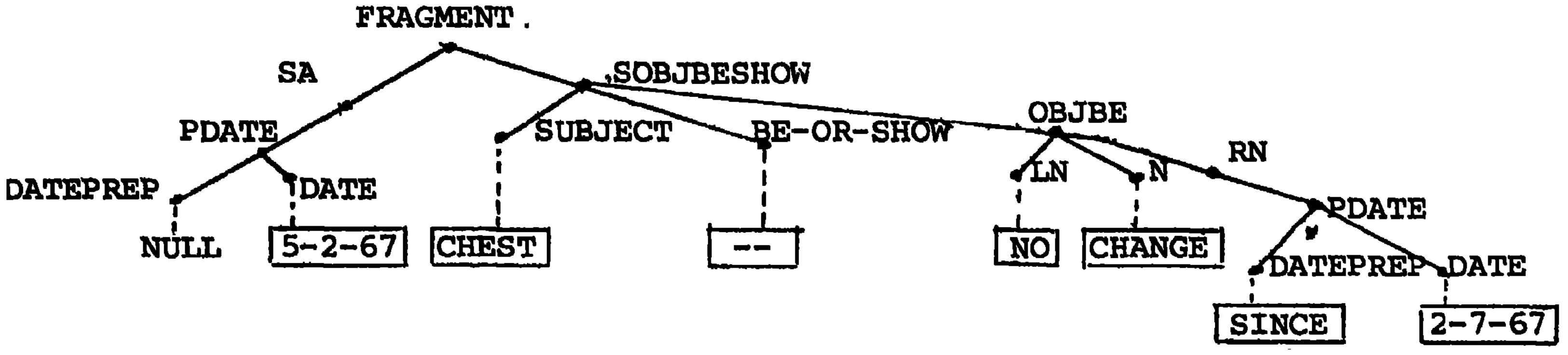
Space permits only a few remarks about these definitions. It was helpful to order the options so that the longer options precede the shorter ones, since some of the shorter options (e.g., NSTG) can have the same form as the first element of the longer ones. This is not required in parsing texts, since in full sentences there is usually no other way of fitting in the remainder of the sentence. Also, in text sentences, many nouns require a preceding determiner, so that compound nouns are not split into separate noun phrases. In this material, determiners are rarely employed, so this constraint cannot be applied. This, combined with verb deletions and the use of commas both in the text and as sentence separators, makes for a great deal of syntactic ambiguity. However, as the next section shows, it was possible to obtain the intended parse as the first output in most cases. This was accomplished without using the subclasses special to the medical material, which are used in a subsequent stage of processing preparatory to information retrieval.

Parsing Results

Parsing output is in the form of a tree, illustrated for a typical fragment in Fig. 1. (Only the nodes mentioned above are shown, plus LN/RN = *l*eft/*r*ight modifiers of *N*oun.) The full power of the parser is better illustrated by the long full sentences; but space does not permit presenting them here.

Fig. 1

Parse tree for FRAGMENT = 5-2-67 chest--no change since 2-7-67



A summary of the parsing results is given in Table 4. Of the total 245 sentences, a correct first parse was obtained for 171 or 69.8%, and a first parse adequate for further processing to obtain an "information format" in 213 cases, or 86.9%. The latter statement brings us to the important question of how these parses are to be used.

TABLE 4. PARSING RESULTS

	Number of Sentences	Percentage
Full sentence	37	15.1
Fragment	205	83.7
Full S + Fragment	3	1.2
TOTAL	245	100.0
1st parse correct	171	69.8
1st parse OK for format	213	86.9
2nd or 3rd parse OK for format	14	6.1
No parse or parses 1-3 not OK for format	17	7.0
TOTAL	245	100.0
Average time for 1st parse	5.158 seconds	

The aim in processing natural language notes and records is to arrive at forms for the data which are suitable for computerized information retrieval. The data structures must not change the meaning. This is why syntactic methods are important. Parsing with an English grammar provides the gross structure of input sentences. (The use of English transformations makes the grammatical analysis more refined.) In each specialized technical area, more specific structure is possible, making use of the restricted word usage characteristic of the discourse in the given subject area.[10]

A second stage of processing of this type is now being applied to the parsed corpus of medical records and will be reported in a subsequent paper. A convenient test of the adequacy of the parsing outputs is therefore whether they can serve as input to this second stage of processing (called formatting). It can be seen in Table 4 that a number of "wrong" parses were still adequate as input to the formatting; the segmentation of the sentence into parts was correct even if the parts were assigned an incorrect syntactic status, e.g., object instead of adjunct. Only when the first parse was not adequate for formatting was the sentence rerun to obtain alternative analyses.

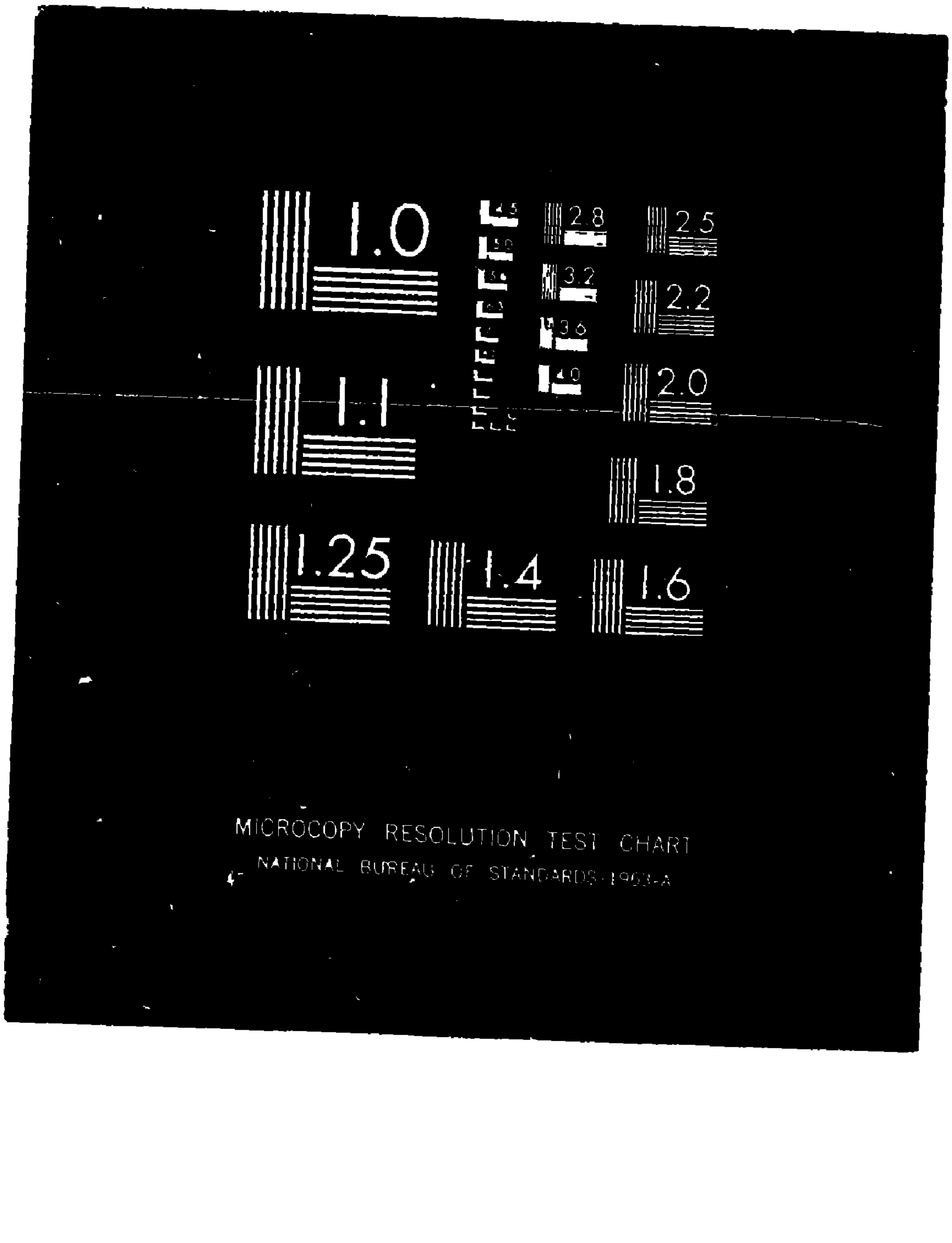
The parsing times are a rough indication of the efficiency of the parsing but two points should be kept in mind. (1) The present LSP system is not a production model, but a research tool, with all that implies. (2) A significant fraction of the input sentences were "no data" types, e.g., <u>None this visit</u>. These word sequences were so limited linguistically that a literal formula could serve to recognize them. The experimental use of such a formula cut down parsing times on the no-data entries from about 1.817 to 0.030. However, this formula was not used in the parsing summarized in Table 4.

ACKNOWLEDGEMENT

This investigation was supported by Public Health Service Research Grant number CA-11531 from the National Cancer Institute.

[10] See Ref. 5 and N. Sager, Syntactic Formatting of Scientific Information, <u>Proc. FJCC</u>, AFIPS Press, Montvale, N. J., 1972.

END


1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6
MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A